Multi-target PHD filtering: proposition of extensions to the multi-sensor case

Emmanuel Delande, Emmanuel Duflos, Dominique Heurguier, Philippe Vanheeghe

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Multi-target PHD filtering: proposition of extensions to the multi-sensor case

Emmanuel DELANDE — Emmanuel DUFLOS — Dominique HEURGUIER — Philippe VANHEEGHE

N° 7337

July 2010

Optimization, Learning and Statistical Methods

Rapport de recherche

ISSN 0249-6399 ISRN INRIA/RR--7337--FR+ENG

INSTITUT NATIONAL
DE RECHERCHE
EN INFORMATIQUE
ET EN AUTOMATIQUE

INRIA

centre de recherche
LILLE - NORD EUROPE

Multi-target PHD filtering: proposition of extensions to the multi-sensor case

Emmanuel DELANDE * , Emmanuel DUFLOS * , Dominique HEURGUIER † , Philippe VANHEEGHE *

Theme : Optimization, Learning and Statistical Methods
Équipe-Projet SequeL

Rapport de recherche n° 7337 — July 2010 — 69 pages

Abstract: Common difficulties in multi-target tracking arise from the fact that the system state and the collection of measurements are unordered and their size evolve randomly through time. The random finite set theory provides a powerful framework to cope with these issues. This document focuses more particularly on the PHD (Probability Hypothesis Density) filter proposed by Mahler.

The first part of this report (up to section 4) is a synthesis of Mahler's work and aims at providing a thorough description of the construction of the single-sensor PHD filter. Then, based on a few leads provided by Mahler, the second part (from section 5) proposes several extensions of this filter to the multi-sensor case.

Key-words: Multi-target/Multi-sensor Tracking, PHD, Random Finite Sets

* LAGIS FRE CNRS 3303 - INRIA Lille Nord Europe (EPI SequeL)
† Thales Communications France

Centre de recherche INRIA Lille – Nord Europe
Parc Scientifique de la Haute Borne
40, avenue Halley, 59650 Villeneuve d'Ascq
Téléphone : +33 3 59 57 78 00 — Télécopie : +33 3 59 57 78 50

Filtre PHD multi-cible : proposition d'extensions au cas multi-capteur

Résumé : Le pistage multi-cible se trouve confronté au double problème suivant : l'état du système et la collection de mesures ne sont pas ordonnés et leurs dimensions varient aléatoirement au cours du temps. Dans ce contexte, l'utilisation des ensembles aléatoires finis apporte un cadre de résolution particulièrement pertinent et ce travail s'intéresse plus particulièrement au filtre PHD (Probability Hypothesis Density) introduit par Mahler.

La première partie de ce rapport (jusqu'à la section 4) est une synthèse des travaux de Mahler et se veut pédagogique : elle reprend en détail la construction du filtre PHD mono-capteur. En se basant sur les éléments de solution proposés par Mahler, la deuxième partie (à partir de la section 5) propose des extensions du filtre au cas multi-capteur.

Mots-clés : Pistage Multi-capteur/Multi-cible, PHD, Ensembles Aléatoires Finis

Contents

1 Introduction

This document deals with the Probability Hypothesis Density (PHD) filter as a solution for multi-target Bayesian filtering. The aim of this document is to describe the setting of the single-sensor/multi-target PHD filter as provided by Mahler, and to propose several paths for an extension to the multi-sensor case. Although the description of the single-sensor case (sections 2, 3 and 4) relies heavily on Mahler's work on the topic ([1], [2]), the author found nonetheless important to describe it thoroughly since it provides ground for the discussion on the multi-sensor case. Pleaste note that this work may be seen as uncomprehensive and may contain mistakes since it mainly reflects the author's modest understanding of Mahler's work on PHD filtering.

Section 2 provides the general framework of multisensor-multitarget Bayesian filtering, that is, it describes the problem to solve. The definition of the PHD and a brief description of the fundamental steps which allows the construction of the PHD-equivalent formulas out of the Bayesian equations are then given in order to provide an insight on the "logical flow" which leads the construction of the single-sensor/multi-target PHD filter.

Section 3 describes some basics on the Finite Set Statistics (FISST) calculus and gives some tools which will be required to "translate" the Bayesian formulas into their PHD-equivalent (section 4). Note that some important mathematical proofs are given in section 8. As other material in these first sections, it is strongly based on Mahler's work ([1], [2]). However, the author did not fully grasp several proofs given in [1] and found it interesting to reformulate them in order improve its understanding on the topic. Moreover, including these proofs allows this document to be as comprehensive as possible.

In section 5, the extension of the single-sensor/multi-target PHD equation to the multi-sensor case are presented. Then, a few leads are discussed in order to simplify the Bayes update equations in a more tractable form. Note that this report is a first version and focuses on theoretical results. Works are in progress to validate the assumptions on the proposed extensions.

2 Multi-sensor/multi-target Bayesian filtering

In the general multisensor-multitarget Bayesian framework, the time update and data update equations at step $k+1$ are given by the following formulas:

$$f_{k+1|k}(X|Z^{(k)}) = \int f_{k+1|k}(X|W) f_{k|k}(W|Z^{(k)}) \delta W \tag{1}$$

$$f_{k+1|k+1}(X|Z^{(k+1)}) = \frac{f_{k+1}(Z_{k+1}|X) f_{k+1|k}(X|Z^{(k)})}{\int f_{k+1}(Z_{k+1}|W) f_{k+1|k}(W|Z^{(k)}) \delta W} \tag{2}$$

where:

- $X = \{x_1, ..., x_n\}$ is a multi-target state, i.e. a finite set of elements x_i defined on the single-target space $\mathcal{X}$; [1]
- $Z_{k+1} = \{z_1, ..., z_m\}$ is the current multi-sensor observation, i.e. a collection of measurements z_i produced at time $k+1$ by all the sensors;
- $Z^{(k)} = \bigcup_{t \leqslant k} Z_t$ is the collection of observations up to time k;
- $f_{k|k}(W|Z^{(k)})$ is the current multi-target posterior density in state W;
- $f_{k+1|k}(X|W)$ is the current multi-target Markov transition density, from state W to state X;
- $f_{k+1}(Z|X)$ is the current multi-sensor/multi-target likelihood function.

Although equations (1) and (2) may seem similar to the classical single-sensor/single-target Bayesians equations, they are generally untractable because of the presence of the *set integrals*. Integrals in (1) and (2) are indeed computed over *multi-target sets* rather than *single-target states*. Because multi-target sets belongs to multi-target space $\mathbb{X} = \bigcup_{n \geqslant 0} \mathcal{X}^n$, to compute every possible set one must compute every possible dimension (i.e number of target) n, and for each n compute every possible n-state collection $(x_1, ..., x_n)$. Likewise, the construction of the multi-target transition function $f_{k+1|k}$ or the multi-sensor/multi-target likelihood function f_{k+1} may be tedious in many tracking problems.

Hence the introduction of the PHD:

Definition 2.1. *([1] p.1154) The Probability Hypothesis Density (PHD) is the density $D_{k|k}(x|Z^{(k)})$ defined on single-target state space $\mathcal{X}$ whose integral $\int_S D_{k|k}(x|Z^{(k)})dx$ on any region $S \subseteq \mathcal{X}$ is the expected number $N_{k|k}(S) = \int |X \cap S| f_{k|k}(X|Z^{(k)}) \delta X$ of targets contained in S.*

Although defined on single-state space $\mathcal{X}$, the PHD encapsulates information on *both* target number and states and therefore provides a nice alternative to cumbersome multitarget posterior $f_{k|k}(X|Z^{(k)})$. Furthermore, should the posterior be approximated as a multitarget Poisson as required later (see section 4.2), Mahler proved ([1], theorem 4 p.1166) that the best Poisson approximation - in an information-theoretic sense - has an intensity equal to its PHD $D_{k|k}(x|Z^{(k)})$.

[1] The state x_i of a target is usually composed of its position, its velocity, etc.

Assuming that propagating the PHD is sufficient enough for an accurate estimation of target number and target states, the challenge is to find the PHD-equivalent of Bayes formulas (1) and (2) in order to propagate densities $D_{k+1|k}(x|Z^{(k)})$ and $D_{k+1|k+1}(x|Z^{(k+1)})$ rather than multi-target densities. Essentially, the construction of the PHD follows from the following points:

1. The multitarget prior $f_{k|k}$ can be seen as a probability distribution f_Ξ of a *random finite set* Ξ;
2. Thanks to Finite Set Statistics (FISST) calculus, f_Ξ and its *multimoment densities* can be constructed as *set derivatives* of its *probability generating density functional* (PGFl) $G_\Xi[h]$;
3. Under certain assumptions - particularly on the target motion model - the PGFl $G_\Xi[h]$ can be constructed explicitly;
4. The PHD $D_{k|k}$ is the first-moment density of the multitarget prior $f_{k|k}$.

Therefore, under the same assumptions, the PHD $D_{k|k}$ can be constructed explicitly. Likewise, the multi-target Bayes equations can be "translated" in explicit PHD-equivalent formulas.

Although the closed-form PHD-equivalent of the time update equation (1) requires no strong assumption on the multitarget posterior (see section 4.1), this is not true for the data update equation (1), which requires the multitarget posterior to be approximately Poisson (see section 4.2). *But*, even with this assumption, the PHD-equivalent remains tractable in the single-sensor case *only.* That is why a few leads are discussed in section 5 in order to simplify the Bayes update equation in the multi-sensor case.

3 Finite set statistics and generalized FISST multi-target calculus

3.1 Multi-target states and finite sets

The multitarget state $X = \{x_1, ..., x_n\}$ can be formulated equivalently by a point process Ξ on space $\mathcal{X}$. If we further assume that the elements x_i are distincts, then the corresponding *simple* point process (or random finite set - RFS) Ξ is equivalently described by the counting measure $N_\Xi(S) = |\Xi \cap S|$ or the Dirac $\delta_\Xi(x) = \sum_{w\in\Xi} \delta_w(x)$. From now on, this assumption will be considered valid so that multi-target sets and the posterior densities can be described using FISST calculus.

A RFS Ξ is characterized by the family of its Janossy densities $j_{\Xi,1}(x_1)$, $j_{\Xi,2}(x_1, x_2)$... (see [6] for a thorough description). These densities are symmetric in all arguments (since elements in Ξ are unordered), vanish whenever two elements are equal (since Ξ is simple), and are jointly normalized:

$$\sum_{n=0}^{\infty} \frac{1}{n!} \int j_{\Xi,n}(x_1, ..., x_n) dx_1 ... dx_n = 1 \tag{3}$$

The multitarget posterior $f_{k|k}(X|Z^{(k)})$ can then be described using the Janossy densities of the corresponding RFS $\Xi_{k|k}$:

$$f_{k|k}(\{x_1, ..., x_n\}|Z^{(k)}) = j_{\Xi_{k|k},n}(x_1, ..., x_n) \tag{4}$$

Because multi-target densities must be described as *simple* RFS in order to use the convenient FISST calculus rules as explained later, an important consequence is that $f_{k|k}(\{x_1, ..., x_n\}|Z^{(k)}) = 0$ whenever two elements x_i, x_j are equal. Whether this might have unwanted consequences on PHD filtering in certain cases (e.g. very close targets in space $\mathcal{X}$) is an interesting question to be answered.

3.2 Probability generating functionals

The probability generating functionals (PGFls) are a generalization of the belief-mass functions that will prove to be convenient tools to compute both multi-target posteriors and PHDs.

Definition 3.1. *([1] p.1161) Let Ξ be a RFS on $\mathcal{X}$ with density f_Ξ and h a real-valued function such that $\forall x \in \mathcal{X},\ 0 \leqslant h(x) \leqslant 1$. The PGFl G_Ξ of Ξ is defined as:*

$$G_\Xi[h] = \int h^X f_\Xi(X) \delta X \quad ^2 \tag{5}$$

As noted in [3], the defintion of the argument function h of PGFl G is simplified here. The gradient derivation of PGFls in Dirac functions (see definition 3.2) requires an extension of h as a sum of Dirac functions; the proper definition is given in [1]. We will ignore this technicality here, because the base functions h and g we will need to derive the PHD equivalent of the time and data update equations are simple real-valued functions as defined above.

Note that the PGFl G_Ξ is the *functional* generalization of the belief-mass *set function* β_Ξ since:

$$\forall S \in \mathcal{X},\ \beta_\Xi(S) = \int 1_S^X f_\Xi(X) \delta X = G_\Xi[1_S] \tag{6}$$

where 1_S is the indicator function on S. Mahler provides the following intuitive interpretation of this functional. If $h(x)$ is seen as the detection probability of a sensor's field of view on a single target space $\mathcal{X}$,

[2] $h^X = \prod_{x\in X} h(x), h^\emptyset = 1$

then $G_\Xi[h]$ is the probability of some RFS Ξ (describing a multi-target state, for example) to be included in the field of view. In that sense, the PGFls can be seen as a generalization of belief-mass functions on fuzzy subsets - with h as their membership function - rather than on "certain states" S.

An important property that will be used to build PGFls is shared with the belief-function. If $\Xi = \Xi_1 \cup ... \cup \Xi_n$ it the union of independent RFSs, then their respective PGFls $G_\Xi, G_{\Xi_1}, ...G_{\Xi_n}$ are linked by the following relation:

$$\forall h, G_\Xi[h] = \prod_i G_{\Xi_i}[h] \tag{7}$$

3.3 Set derivatives and functional derivatives

In a similar fashion than the gradient derivative of real functions $G(x)$ on $\mathcal{X}$ are defined, the gradient derivative of a PGFl $G[h]$ in function direction g is given by:

$$\frac{\partial G}{\partial g}[h] = \lim_{\epsilon \to 0} \frac{G[h + \epsilon.g] - G[h]}{\epsilon} \tag{8}$$

The PGFls being the "functional conterpart" (although expanded) of the "set" belief-mass functions, Mahler showed using set derivatives rules that *gradient derivatives* on PGFls are equivalent to *set derivatives* on belief-mass functions. That is, for all multi-target set $X = \{x_1, ..., x_n\}$ for all subset $S \subseteq \mathcal{X}$:

$$\frac{\delta^n \beta_\Xi}{\delta x_1 ... \delta x_n}(S) = \frac{\partial^n G_\Xi}{\partial \delta_{x_1} ... \partial \delta_{x_n}}[1_S] \tag{9}$$

That is, the set derivative in the singleton $\{x\}$ is the conterpart of the gradient derivative in Dirac delta function δ_x (also called functional derivative in x). Hence the following simplified notation of gradient derivatives:

Definition 3.2. *([1] p.1162) The functional derivatives of PGFl $G[h]$ in distinct points $x_1, ..., x_n$ are:*

$$\frac{\delta^0 G}{\delta x^0}[h] = G[h] \tag{10}$$

$$\frac{\delta G}{\delta x_1}[h] = \frac{\partial G}{\partial \delta_{x_1}}[h] \tag{11}$$

$$\frac{\delta^n G}{\delta x_1 ... \delta x_n}[h] = \frac{\partial^n G}{\partial \delta_{x_1} ... \partial \delta_{x_n}}[h] \tag{12}$$

3.4 Some essential properties of functional derivatives

Let $G[h] = h(x_1)$ for some $x_1 \in \mathcal{X}$, then:

$$\frac{\delta G}{\delta x}[h] = \delta_x(x_1) \tag{13}$$

The proof is given in section 8.1.

Let $G[h] = \int h(x) f(x) dx$, then:

$$\frac{\delta G}{\delta x}[h] = f(x) \tag{14}$$

The proof is given in section 8.2.

Proposition 3.1. *Let Ξ be a RFS with density f_Ξ and PGFl G_Ξ, $X = \{x_1, ..., x_n\}$ a RFS of distinct points of $\mathcal{X}$, then:*

$$f_\Xi(X) = \frac{\delta^n G_\Xi}{\delta x_1 ... \delta x_n}[0] \tag{15}$$

The proof is given in section 8.3.

3.5 Multitarget moment densities

Definition 3.3. *([1] p.1162) Let Ξ be a RFS on $\mathcal{X}$ with density f_Ξ. Its moment density D_Ξ is defined as:*

$$D_\Xi(X) = \int f_\Xi(X \cup W)\delta W \tag{16}$$

The function of n variables $D_\Xi(\{x_1, ..., x_n\})$ is the moment of order n (or n-th moment) of Ξ.

The moment density $D_\Xi(X)$ of a given multi-target set $X = \{x_1, ..., x_n\}$ can be seen as the likelihood that a realisation of the RFS Ξ contains *at least* the points $x_1, ..., x_n$. Therefore, if the state space $\mathcal{X}$ is a single-target space, $D_\Xi(X)$ can be seen as the likelihood that there are *at least* n targets, with distinct states $x_1, ..., x_n$.

As for the density f_Ξ, the moment density D_Ξ can be computed as a set derivative of the PGFl.

Theorem 3.1. *([1] p.1162) Let Ξ be a RFS with density f_Ξ, moment density D_Ξ and PGFl G_Ξ, $X = \{x_1 ... x_n\}$ a collection of distinct points of $\mathcal{X}$, then:*

$$D_\Xi(X) = \frac{\delta^n G_\Xi}{\delta x_1 ... \delta x_n}[1] \tag{17}$$

The proof is given in section 8.4.

The following calculus rule is an example of the use of the first moment in order to compute set integrals.

Proposition 3.2. *([1] p.1163) Let Ξ be a RFS with density f_Ξ and moment density D_Ξ and PGFl G_Ξ, and h a real-valued function on $\mathcal{X}$. Then:*

$$\int h_Y f_\Xi(Y)\delta Y = \int h(y) D_\Xi(\{y\}) dy \quad ^3 \tag{18}$$

In particular, setting $h = \delta_x$ gives ([1] p.1162) :

$$\forall x \in \mathcal{X}, \quad D_\Xi(\{x\}) = \int \delta_Y(x) f_\Xi(Y)\delta Y \quad ^4 \tag{19}$$

The proof can be found in section 8.5.

Another important result involves PGFls and functional tranformation:

Proposition 3.3. *([1] p.1163) Let $h \to \Phi[h]$ be a functional transformation such that $\Phi[1] = 1$. Let G_Ξ be a PGFl with corresponding moment density D_Ξ, and G the functional defined by $G[h] = G_\Xi[\Phi[h]]$ for all h, with corresponding moment density D. Then:*

$$D(\{x\}) = \int D_w(\{x\}) D_\Xi(\{w\}) dw \tag{20}$$

where D_w is the corresponding moment density of the PGFl Φ_w defined by $\Phi_w[h] = \Phi[h](w)$.

[3] $h_X = \sum_{x \in X} h(x), h_\emptyset = 0$
[4] $\delta_Y(x) = \sum_{y \in Y} \delta_y(x), \delta_\emptyset = 0$

The proof is given in section 8.6.

The fundamental result for later is that the PHD is the first moment density:

Theorem 3.2. *([1] p.1163) Let $\Xi_{k|k}$ be a multitarget RFS with density $f_{k|k}(X|Z^{(k)})$ and moment density $D_{k|k}(X|Z^{(k)})$. Then, for any region $S \subseteq \mathcal{X}$:*

$$\int_S D_{k|k}(\{x\}|Z^{(k)})dx = \mathbb{E}(|\Xi_{k|k} \cap S|) \tag{21}$$

That is, the first-moment density equals the PHD almost everywhere:

$$D_{k|k}(\{x\}|Z^{(k)}) = D_{k|k}(x|Z^{(k)}) \tag{22}$$

The proof is given in section 8.7.

Equation (21) is fundamental because it provides the PHD with a traditionnal statistical interpretation. Being the first moment density, the PHD can be seen as the "expectation" in the multitarget space. Indeed, the set integral $\int X.f_{k|k}(X|Z^{(k)})\delta X$ does not exist since the sum operator has no sense in the class of finite sets, hence the absence of a "conventional" expectation operator.

Note also that the combination of results (17) and (22) allows the computation of the PHD as a set derivative, since:

$$D_\Xi(x) = D_\Xi(\{x\}) = \frac{\delta G_\Xi}{\delta x}[1] \tag{23}$$

Since the PHD $D_{k|k}(x|Z^{(k)})$ is merely the first-moment of the multi-target density $f_{k|k}(X|Z)$, propagating the PHD rather than the multi-target density reduces the information available on targets. Although this cost is necessary to transform general Bayes equations in their more tractable PHD-equivalents, one must keep it in mind for the design of a practical filter.

4 Single-sensor/multi-target PHD filtering

4.1 Time update equation

The challenge is to find the conterpart of the set-based Bayes time update equation (1), repeated here:

$$f_{k+1|k}(X|Z^{(k)}) = \int f_{k+1|k}(X|W)f_{k|k}(W|Z^{(k)})\delta W$$

First, some assumptions on the target model must be made in order to shape the corresponding PGFls. It is assumed that, at time k:

- Each existing target i, with state x_k^i, dies with a probability $1 - p_s(x_i^k)$; [5]
- If it survives, target x_k^i evolves towards state x_{k+1}^i with probability distribution $f_{k+1|k}^E(.|x_k^i)$;
- For each existing target with state x_k^i, a collection of spawned targets Y_{k+1}^i is created whose probability distribution is $f_{k+1|k}^S(.|x_k^i)$;
- A collection of spontaneous birth targets Y_{k+1} is created whose probability distribution is $f_{k+1|k}^B(.)$;
- The evolution, spawning and spontaneous processes are statistically indendent conditionnaly on the multitarget state X_k of existing targets.

Note that the transition $f_{k+1|k}^E(.|x_k^i)$ is a *state* function, but the spawning $f_{k+1|k}^S(.|x_k^i)$ and the birth $f_{k+1|k}^B(.)$ processes are still *set* function. The above assumptions on the target transition model are fundamental because they allow the "decoupling" of the *set* transition function $f_{k+1|k}(.|W)$ in a much simpler *state* function $f_{k+1|k}^E(.|x_k^i)$. Therefore, with no additional assumptions on spawning or birth processes, the sole part of the PGFl of the multi-target state RFS $\Xi_{k+1|k}$ which is contructed explicitly is the transition part and we have:

Proposition 4.1. *([1] p.1164) Conditionnaly on previous multi-target set X, the multi-target RFS $\Xi_{k+1|k}$ with density $f_{k+1|k}(Y|X)$ has a PGFl $G_{k+1|k}[h|X]$ equal to:*

$$G_{k+1|k}[h|X] = (1 - p_s + p_s.p_h^E)^X.(p_h^S)^X.p_h^B \tag{24}$$

where:

- *$p_h^E(.)$ is the function $p_h^E(x) = \int h(y)f_{k+1|k}^E(y|x)dy$;*
- *$p_h^S(x)$ is the PGFl of the spawning multi-target RFS $\Xi_{k+1|k}^S(x)$;*
- *p_h^B is the PGFl of the spontaneous birth multi-target RFS $\Xi_{k+1|k}^B$.*

The proof is given in section 8.8.

[5] $p_s(.) = p_{s,k+1}(.)$, time subscripts are omitted for simplicity's sake.

From the construction of the PGFl, the time update formula follows:

Theorem 4.1. *([1] p.1167) The time update equation is given by:*

$$D_{k+1|k}(x) = b^B_{k+1|k}(x) + \int \Big(p_s(w) f^E_{k+1|k}(x|w) + b^S_{k+1|k}(x|w)\Big) D_{k|k}(w)dw \tag{25}$$

where:

- $D_{k+1|k}(x) = D_{k+1|k}(x|Z^{(k)})$ *is the PHD of the time updated multi-target RFS* $\Xi_{k+1|k}$*;*
- $D_{k|k}(x) = D_{k|k}(x|Z^{(k)})$ *is the PHD of the posterior multi-target RFS* $\Xi_{k|k}$*;* [6]
- $b^S_{k+1|k}(x|w)$ *is the PHD of the spawning multi-target RFS* $\Xi^S_{k+1|k}(w)$*;*
- $b^B_{k+1|k}(x)$ *is the PHD of the spontaneous birth multi-target RFS* $\Xi^B_{k+1|k}$.

The proof is given in 8.9.

Theorem 4.1 allows (in theory) the computation of the time updated expected number of targets $N_{k+1|k}$ since, according to definition 2.1:

$$\begin{aligned} N_{k+1|k} &= \int D_{k+1|k}(x)dx \\ &= \int b^B_{k+1|k}(x)dx + \int \left[\left(p_s(w) + \int b^S_{k+1|k}(x|w)dx\right) D_{k|k}(w)\right] dw \end{aligned} \tag{26}$$

However, depending on the shape of the spawning and spontaneous birth processes, the computation of integrals $\int b^B_{k+1|k}(x)dx$ and $\int b^S_{k+1|k}(x|w)dx$ for all w may be tricky. In order to overcome this difficulty, the *set* functions of the spawning and spontaneous birth processes can be "decoupled" in *state* functions so that their PGFl, and henceforth their PHD, can be constructed explicitly. As for the transition function, this requires additionnal assumptions. It is common to assume than spawning and spontaneous birth processes are Poisson, i.e.:

$$f^S_{k+1|k}(Y|x) = \exp(-\lambda^S_{k+1|k}(x)) \prod_{y\in Y} \lambda^S_{k+1|k}(x) s^S_{k+1|k}(y|x) \tag{27}$$

$$f^B_{k+1|k}(Y) = \exp(-\lambda^B_{k+1|k}) \prod_{y\in Y} \lambda^B_{k+1|k} s^B_{k+1|k}(y) \tag{28}$$

where:

- $\forall x$, $I^S(y|x) = \lambda^S_{k+1|k}(x) s^S_{k+1|k}(y|x)$ with $\lambda^S_{k+1|k}(x) \geqslant 0$ and $\int s^S_{k+1|k}(y|x)dy = 1$ is the *intensity* of the spawning process in y originating from target in x;
- $I^B(y) = \lambda^B_{k+1|k} s^B_{k+1|k}(y)$ with $\lambda^B_{k+1|k} \geqslant 0$ and $\int s^B_{k+1|k}(y)dy = 1$ is the *intensity* of the spontaneous birth process in y.

[6] $D_{k|k}(x)$ is stated as "posterior" since it is the density which results from the Bayes update following the previous observation.

The following proposition is a useful tool when dealing with Poisson processes:

Proposition 4.2. *([1] p.1164) Let Ξ be a RFS whose density $f_\Xi(X)$ is Poisson with intensity $I(x)$ and parameter $\lambda = \int I(x)dx$. Then:*

- *Its moment density D_Ξ is $D_\Xi(X) = \prod_{x \in X} I(x)$;*
- *Its PGFl G_Ξ is $G_\Xi[h] = \exp(I[h] - \lambda)$.*

where $I[h]$ is the functional $I[h] = \int h(x)I(x)dx$.

The proof is given in 8.10.

Since the PHD is the first-moment density (see (22)) and therefore the intensity (see proposition 4.2 above), then the Poisson assumption immediately leads to the following simplified versions of theorem 4.1 and equality (26):

Corollary 4.1. *([1] P.1167) If the spawning and spontaneous birth processes are Poisson with respective intensities $I^S(y|x) = \lambda^S_{k+1|k}(x) s^S_{k+1|k}(y|x)$ and $I^B(y) = \lambda^B_{k+1|k} s^B_{k+1|k}(y)$, the time update equation is given by:*

$$D_{k+1|k}(x) = \lambda^B_{k+1|k} s^B_{k+1|k}(x) + \int \left(p_s(w) f^E_{k+1|k}(x|w) + \lambda^S_{k+1|k}(w) s^S_{k+1|k}(x|w) \right) D_{k|k}(w)dw \tag{29}$$

Furthermore, the time updated expected number of targets $N_{k+1|k}$ is given by:

$$N_{k+1|k} = \lambda^B_{k+1|k} + \int \left(p_s(w) + \lambda^S_{k+1|k}(w) \right) D_{k|k}(w)dw \tag{30}$$

From now on, we will assume spawning and spontaneous birth processes to be Poisson so that results from corollary 4.1 are valid.

4.2 Single-sensor data update equation

The challenge is to find the conterpart of the set-based data update equation (2), repeated here:

$$f_{k+1|k+1}(X|Z^{(k+1)}) = \frac{f_{k+1}(Z_{k+1}|X) f_{k+1|k}(X|Z^{(k)})}{\int f_{k+1}(Z_{k+1}|W) f_{k+1|k}(W|Z^{(k)}) \delta W}$$

First, some assumptions on the observation model must be made in order to shape the corresponding PGFls. It is assumed that, following target transition between time steps k and $k+1$:

- Each existing target i, with state x^i_{k+1}, is detected with probability $p_d(x^i_{k+1})$; [7]
- If detected, target i produces a *single* measurement $z \in \mathcal{Z}$ with probability distribution $f^O_{k+1}(z|x^i_{k+1}) = L_z(x^i_{k+1})$ (L_z is the vraisemblance function) ;
- A collection of false alarms are created, who is assumed Poisson with parameter λ and intensity $\lambda.c(.)$;
- The observation processes on each target are statistically independent, conditionnaly on the multi-target set X_{k+1} of existing targets.

[7] $p_d(.) = p_{d,k+1}(.)$, time subscripts are omitted for simplicity's sake.

Proposition 4.3. *([1] p.1168) Conditionnaly on previous multi-target set X, the single-sensor/multi-target observation RFS $\Sigma_{k+1|k}$ with density $f_{k+1}(Z_{k+1}|X)$ has a PGFl $G_{k+1}[g|X]$ equal to:*

$$G_{k+1}[g|X] = (1 - p_d + p_d.p_g^O)^X.e^{\lambda.c[g]-\lambda} \tag{31}$$

where:

- $p_g^O(.)$ *is the function* $p_g^O(x) = \int g(z)f_{k+1}^O(z|x)dz$*;*
- $c[.]$ *is the false alarm functional* $c[g] = \int g(z)c(z)dz$*.*

The proof is given in 8.11.

In order to find the PHD-equivalent of the set-based data update, an assumption on the time updated PHD (i.e. the density $D_{k+1|k}$ resulting from corollary 4.1) is necessary (see below). From now on, it will be assumed that time updated distribution $f_{k+1|k}(X|Z^{(k)})$ is approximately Poisson with intensity μs and parameter μ [8]. The following definition introduces the "cross-terms". Even though these terms were not coined by Mahler, they will be introduced since they will play an important role in the extension to the multi-sensor case (see section 5).

Definition 4.1. *Let g be a real-valued function defined on observation space $\mathcal{Z}$ such that $\forall z \in \mathcal{Z}^{[j]}$, $0 \leqslant g^{[j]}(z) \leqslant 1$. Let h be a real-valued function defined on state space $\mathcal{X}$ such that $\forall x \in \mathcal{X}$, $0 \leqslant h(x) \leqslant 1$. Assume that the time updated distribution $f_{k+1|k}(X|Z^{(k)})$ is approximately Poisson with intensity μs and parameter μ. Then, the generic single-sensor cross-term is the functional defined by:*

$$\beta[g,h] = \lambda c[g] - \lambda + \mu s\left[h(1 - p_d + p_d p_g^O)\right] - \mu \tag{32}$$

where:

- p_g^O *is the function* $p_g^O(x) = \int g(z)f_{k+1}^O(z|x)dz$*;*
- c *is the false alarm functional* $c[g] = \int g(z)c(z)dz$*.*

As for the PGFLs, g and h can be seen as fuzzy membership functions in observation space $\mathcal{Z}$ and target state space $\mathcal{X}$ respectively. When differentiated in a single point of state space x and/or observation space z, the *generic* cross-term becomes *set* in x and/or z:

Definition 4.2. *Let $\beta[g,h]$ be the generic single-sensor cross-term. Let $x \in \mathcal{X}$ be any point in target space. The single-sensor cross-term set in point x is defined by:*

$$\beta[g,\delta_x] = \frac{\delta}{\delta x}\beta[g,h]$$

Let $z \in Z_{k+1}$ be one of the current measurements, where Z_{k+1} is assumed non empty. The single-sensor cross-term set in z is defined by:

$$\beta[\delta_z,h] = \frac{\delta}{\delta z}\beta[g,h]$$

The single-sensor cross-term set in x and z is defined by:

$$\beta[\delta_z,\delta_x] = \frac{\delta}{\delta z}\beta[g,\delta_x] (= \frac{\delta}{\delta x}\beta[\delta_z,h])$$

[8] $\mu s(.) = \mu_{k+1|k}s_{k+1|k}(.)$, time subscripts are omitted for simplicity's sake

The following proposition gives an explicit expression of the set cross-terms and provides ground for their intuitive interpretation:

Proposition 4.4. *With the same notations as in definition 4.2, we have:*

$$\begin{aligned}\beta[g,\delta_x] &= \mu(1-p_d(x)+p_d(x)p_g^O(x))s(x)\\ \beta[\delta_z,h] &= \lambda c(z)+\mu s[hp_dL_z]\\ \beta[\delta_z,\delta_x] &= \mu p_d(x)L_z(x)s(x)\end{aligned}$$

In particular, setting $g=0, h=1$ gives:

$$\beta[0,\delta_x] = \mu(1-p_d(x))s(x) \tag{33}$$
$$\beta[\delta_z,1] = \lambda c(z)+\mu s[p_dL_z] \tag{34}$$
$$\beta[\delta_z,\delta_x] = \mu p_d(x)L_z(x)s(x) \tag{35}$$

The proof is given in 8.12. These cross-terms can therefore be seen as the likelihood that:

- $\beta[0,\delta_x]$: a target is in state x *and* is undetected by the current observation;
- $\beta[\delta_z,\delta_x]$: a target is in state x, is detected *and* produces measurement z;
- $\beta[\delta_z,1]$: a measurement is produced in point z.

Since $\beta[\delta_z,1]$ is the likelihood that a measurement is produced in point z, it covers both possible events that z is a false alarm ("$\lambda c(z)$") *or* z is produced by an underlying target ("$\mu s[p_dL_z]$", which can be seen as an "expectation" over state space). Note also that, according to the results of the proposition above, a cross-term which is set in *several* observation points (e.g. $\frac{\delta^2}{\delta z_2\delta z_1}\beta[g,h]=\frac{\delta}{\delta z_2}\beta[\delta_{z_1},h]$) and/or in *several* state points (e.g. $\frac{\delta^2}{\delta x_2\delta x_1}\beta[g,h]=\frac{\delta}{\delta x_2}\beta[g,\delta_{x_1}]$) equals to zero. Indeed, under the observation model, the likelihood that several targets produce a single measurement and, alternatively, the likelihood that several measurements are produced by a single target are null.

Finally, the single-sensor data update formula can be constructed with derivatives of the generic single-sensor terms and constructed explicitely as follows:

Theorem 4.2. *([1] p.1168) Let $Z_{k+1}=\{z_1,...,z_m\}$ be the set of current measurements. Assuming that the time updated distribution $f_{k+1|k}(X|Z^{(k)})$ is approximately Poisson with parameter μ and intensity $\mu s(x)$, the data update equation is given by:*

$$D_{k+1|k+1}(x)=\frac{\left[\frac{\delta}{\delta x}\left(\frac{\delta^m}{\delta z_1\ldots\delta z_m}e^{\beta[g,h]}\right)\right]_{g=0,h=1}}{\left[\frac{\delta^m}{\delta z_1\ldots\delta z_m}e^{\beta[g,h]}\right]_{g=0,h=1}}=\left(1-p_d(x)+\sum_{z\in Z_{k+1}}\frac{p_d(x)L_z(x)}{\lambda c(z)+D_{k+1|k}[p_dL_z]}\right).D_{k+1|k}(x) \tag{36}$$

where:

- $D_{k+1|k+1}(x)=D_{k+1|k+1}(x|Z^{(k+1)})$ *is the PHD of the data updated multi-target RFS* $\Xi_{k+1|k+1}$*;*
- $D_{k+1|k}(x)=D_{k+1|k}(x|Z^{(k)})$ *is the PHD of the time updated multi-target RFS* $\Xi_{k+1|k}$*;*
- $D_{k+1|k}[h]$ *is the functional* $D_{k+1|k}[h]=\int h(x)D_{k+1|k}(x)dx$*.*

The proof is given in 8.13.

5 Extension to the multi-sensor case

As it can be seen in proof 8.13, the single-sensor data update equation is tractable because the set derivatives on the single-sensor cross-terms are easy to compute. Although it is quite straightforward to expand the cross-terms to multi-sensor ones and to write the multi-sensor data update equation as set derivatives of these expanded cross-terms, finding a closed-form expression of the data updated multi-sensor PHD (as an equivalent to theorem 4.2) is more challenging. Mahler provided [4] a combinational expression for two sensors, in this section we will extend these results to the multi-sensor case with our own expression based on multi-sensor cross-terms. Fortunately, our results when the sensor number is $N = 2$ match Mahler's.
Note that the time update equation does not involve the observation process (see theorem 4.1) and is therefore identical in the single-sensor and multi-sensor cases.

5.1 Multi-sensor data update equation: derivative form

Assume that, following target transition between time steps k and $k+1$, each sensor $j \in [1\ N]$ produces measurements *independently* of the others according to the observation model described in section 4.2:

- Each existing target i, with state x^i_{k+1}, is detected with probability $p^{[j]}_d(x^i_{k+1})$[9];
- If detected, target i produces a *single* measurement $z \in \mathcal{Z}^{[j]}$ with probability distribution $f^{O,[j]}_{k+1}(z|x^i_{k+1}) = L^{[j]}_z(x^i_{k+1})$;
- A collection of false alarms are created, which is assumed Poisson with parameter $\lambda^{[j]}$ and intensity $\lambda^{[j]}c^{[j]}(.)$;
- The observation processes on each target are statistically independent, conditionnaly on the multi-target set X_{k+1} of existing targets.

Further assume that the time updated distribution $f_{k+1|k}(X|Z^{(k)})$ is approximately Poisson with parameter μ and intensity $\mu.s(x)$. Then, expanding the cross-terms defined in the proof of the single-sensor data update equation (see proof in section 8.13) gives:

Definition 5.1. *For all $j \in [1\ N]$, let $g^{[j]}$ be a real-valued function defined on observation space $\mathcal{Z}^{[j]}$ such that $\forall z \in \mathcal{Z}^{[j]}$, $0 \leqslant g^{[j]}(z) \leqslant 1$. Let h be a real-valued function defined on state space $\mathcal{X}$ such that $\forall x \in \mathcal{X}$, $0 \leqslant h(x) \leqslant 1$. The generic multi-sensor cross-term $\beta[g^{[1]}, ..., g^{[N]}, h]$ is the functional defined by:*

$$\beta[g^{[1]}, ..., g^{[N]}, h] = \sum_{j=1}^{N} (\lambda^{[j]} c^{[j]}[g^{[j]}] - \lambda^{[j]}) + \mu s\left[h \prod_{j=1}^{N} \left(1 - p^{[j]}_d + p^{[j]}_d p^{O,[j]}_{g^{[j]}}\right)\right] - \mu \tag{37}$$

where:

- *$p^{O,[j]}_{g^{[j]}}(.)$ is the function $p^{O,[j]}_{g^{[j]}}(x) = \int g^{[j]}(z) f^{O,[j]}_{k+1}(z|x)dz$;*
- *$c^{[j]}[.]$ is the false alarm functional $c^{[j]}[g^{[j]}] = \int g^{[j]}(z)c^{[j]}(z)dz$.*

As for the single-sensor case, $g^{[1]}, ..., g^{[N]}$ and h can be seen as fuzzy membership functions in observation space $\mathcal{Z}^{[1]}, ..., \mathcal{Z}^{[N]}$ and target state space $\mathcal{X}$ respectively. Since any target may be detected by up to N sensors and hence may produce up to N different measurements, one in each observation spaces $\mathcal{Z}^{[i]}$. Therefore, the *generic* multi-sensor cross-term can be *set* in a single state x and/or any combination of measurements, at most one per sensor:

[9] $p^{[j]}_d(.) = p^{[j]}_{d,k+1}(.)$, time subscripts are omitted for simplicity's sake

Definition 5.2. *Let $\beta[g^{[1]},...,g^{[N]},h]$ be the generic cross-term. Let $x \in \mathcal{X}$ be any element in target space. The multi-sensor cross-term set in point x is defined by:*

$$\beta[g^{[1]},...,g^{[N]},\delta_x] = \frac{\delta}{\delta x}\beta[g^{[1]},...,g^{[N]},h]$$

Let $1 \leqslant M \leqslant N$ and $t_1,...t_M$ be the increasing indexes in $[1\ N]$ of any M distinct sensors. Let $z^{[t_j]} \in Z_{k+1}^{[t_j]}$ be one of the current measurements of sensor t_j, where $Z_{k+1}^{[t_j]}$ is assumed non empty. The multi-sensor cross-term set in $z^{[t_1]},...,z^{[t_M]}$ is defined by:

$$\beta[g^{[1]},...,\delta_{z^{[t_1]}},...,\delta_{z^{[t_M]}},...,g^{[N]},h] = \frac{\delta^M}{\delta z^{[t_1]}...\delta z^{[t_M]}}\beta[g^{[1]},...,g^{[N]},h]$$

The multi-sensor cross-term set in measurements $z^{[t_1]},...,z^{[t_M]}$ and target state x is defined by:

$$\begin{aligned}\beta[g^{[1]},...,\delta_{z^{[t_1]}},...,\delta_{z^{[t_M]}},...,g^{[N]},\delta_x] &= \frac{\delta^M}{\delta z^{[t_1]}...\delta z^{[t_M]}}\beta[g^{[1]},...,g^{[N]},\delta_x] \\ &\left(= \frac{\delta}{\delta x}\beta[g^{[1]},...,\delta_{z^{[t_1]}},...,\delta_{z^{[t_M]}},...,g^{[N]},h]\right)\end{aligned}$$

The following proposition gives an explicit expression of the set cross-terms and provides ground for their intuitive interpretation:

Proposition 5.1. *With the same notation as in definition 5.2, we have:*

$$\beta[g^{[1]},...,g^{[N]},\delta_x] = \mu\prod_{j=1}^{N}\left(1 - p_d^{[j]}(x) + p_d^{[j]}(x)p_{g^{[j]}}^{O,[j]}(x)\right)s(x)$$

$$\beta[g^{[1]},...,\delta_{z^{[t_1]}},...,\delta_{z^{[t_M]}},...,g^{[N]},h] = \begin{cases}\lambda^{[t_1]}c^{[t_1]}(z^{[t_1]}) + \mu s\left[h\left(p_d^{[t_1]}L_{z^{[t_1]}}^{[t_1]}\right)\prod_{j\neq t_1}\left(1 - p_d^{[j]} + p_d^{[j]}p_{g^{[j]}}^{O,[j]}\right)\right] & (M=1) \\ \mu s\left[h\prod_{i=1}^{M}\left(p_d^{[t_i]}L_{z^{[t_i]}}^{[t_i]}\right)\prod_{j\notin\{t_1,...,t_M\}}\left(1 - p_d^{[j]} + p_d^{[j]}p_{g^{[j]}}^{O,[j]}\right)\right] & (M>1)\end{cases}$$

$$\beta[g^{[1]},...,\delta_{z^{[t_1]}},...,\delta_{z^{[t_M]}},...,g^{[N]},\delta_x] = \mu\prod_{i=1}^{M}\left(p_d^{[t_i]}(x)L_{z^{[t_i]}}^{[t_i]}(x)\right)\prod_{j\notin\{t_1,...,t_M\}}\left(1 - p_d^{[j]}(x) + p_d^{[j]}(x)p_{g^{[j]}}^{O,[j]}(x)\right)s(x)$$

In particular, setting $g^{[1]} = 0,...,g^{[N]} = 0, h = 1$ gives:

$$\beta[0,...,0,\delta_x] = \mu\prod_{j=1}^{N}\left(1 - p_d^{[j]}(x)\right)s(x) \tag{38}$$

$$\beta[0,...,\delta_{z^{[t_1]}},...,\delta_{z^{[t_M]}},...,0,1] = \begin{cases}\lambda^{[t_1]}c^{[t_1]}(z^{[t_1]}) + \mu s\left[\left(p_d^{[t_1]}L_{z^{[t_1]}}^{[t_1]}\right)\prod_{j\neq t_1}\left(1 - p_d^{[j]}\right)\right] & (M=1) \\ \mu s\left[\prod_{i=1}^{M}\left(p_d^{[t_i]}L_{z^{[t_i]}}^{[t_i]}\right)\prod_{j\notin\{t_1,...,t_M\}}\left(1 - p_d^{[j]}\right)\right] & (M>1)\end{cases} \tag{39}$$

$$\beta[0,...,\delta_{z^{[t_1]}},...,\delta_{z^{[t_M]}},...,0,\delta_x] = \mu\prod_{i=1}^{M}\left(p_d^{[t_i]}(x)L_{z^{[t_i]}}^{[t_i]}(x)\right)\prod_{j\notin\{t_1,...,t_M\}}\left(1 - p_d^{[j]}(x)\right)s(x) \tag{40}$$

The proof is given in 8.14. Once simplified, these cross-terms have an intuitive interpretation and can be seen as the likelihood that:

- $\beta[0, ..., 0, \delta_x]$: a target is in state x *and* is undetected by the current observation;
- $\beta[0, ..., \delta_{z^{[t_1]}}, ..., \delta_{z^{[t_M]}}, ..., 0, 1]$: measurements are produced in points $z^{[t_j]}, j \in [1\ M]$ by a *single* source;
- $\beta[0, ..., \delta_{z^{[t_1]}}, ..., \delta_{z^{[t_M]}}, ..., 0, \delta_x]$: a target is in state x, is detected by sensors $t_j, j \in [1\ M]$ *only and* produces measurements $z^{[t_j]}, j \in [1\ M]$.

Note that the false alarm term $\lambda^{[i]} c^{[i]}(z^{[i]})$ vanished in $\beta[0, ..., \delta_{z^{[t_1]}}, ..., \delta_{z^{[t_M]}}, ..., 0, 1]$ whenever $M > 1$. Indeed, cross-terms quantify the likelihood of the "link" between several points, at most one in state space (hence only one fuzzy membership function on $\mathcal{X}$, namely h), at most one in each of the observation spaces (hence one fuzzy membership function per observation space $\mathcal{Z}^{[i]}$, namely $g^{[i]}$). Hence, each cross-term quantifies the likelihood that the measurements are produced by a single source. If there is only one measurement (i.e. $M = 1$), the measurement can be either produced by an underlying target or by the false alarm process; if there are at least two measurements (i.e. $M > 1$), all the measurements are produced by a *single* target.

As it will be seen later, the case where several measurements from different sensors are produced by false alarm processes is covered by a combination of cross-terms. For example, the product $\beta[\delta_{z^{[1]}}, 0, 1].\beta[0, \delta_{z^{[2]}}, 1]$ stands for the likelihood that measurements $z^{[1]}$ and $z^{[2]}$ are produced by *different* sources, each one can be produced by an underlying target or by a false alarm process, *provided that* the (eventual) underlying targets are different. On the other hand, the cross-term $\beta[\delta_{z^{[1]}}, \delta_{z^{[2]}}, 1]$ stands for the likelihood that measurements $z^{[1]}$ and $z^{[2]}$ are produced by a *single* source, which is necessarily a *single* underlying target.

As for the single-sensor case, a cross-term which is set in *several* observation points from the same observation space (e.g. $\frac{\delta^2}{\delta z_2^{[1]} \delta z_1^{[1]}} \beta[g^{[1]}, g^{[2]}, h] = \frac{\delta}{\delta z_2^{[1]}} \beta[\delta_{z_1^{[1]}}, g^{[2]}, h]$) and/or in *several* state points (e.g. $\frac{\delta^2}{\delta x_2 \delta x_1} \beta[g^{[1]}, g^{[2]}, h] = \frac{\delta}{\delta x_2} \beta[g^{[1]}, g^{[2]}, \delta_{x_1}]$) equals to zero. Indeed, under the observation model, the likelihood that several targets produce a single measurement and, alternatively, the likelihood that several measurements are produced by a single target are null.

The multi-sensor data update equation can then be constructed as the ratio of set derivatives from the generic multi-sensor cross-term:

Proposition 5.2. *For all sensor $j \in [1\ N]$, let $Z^{[j]}_{k+1} = \{z^{[j]}_1, ..., z^{[j]}_{m[j]}\}$ be the set of current measurements produced by sensor j. Assuming that the time updated distribution $f_{k+1|k}(X|Z^{(k)})$ is approximately Poisson with parameter μ and intensity $\mu s(x)$, the data update equation is given by:*

$$D_{k+1|k+1}(x) = \frac{\left[\frac{\delta}{\delta x}\left(\frac{\delta^{m[1]+\ldots+m[N]}}{\delta z^{[1]}_1 \ldots \delta z^{[1]}_{m[1]} \ldots \delta z^{[N]}_1 \ldots \delta z^{[N]}_{m[N]}} e^{\beta[g^{[1]},\ldots,g^{[N]},h]}\right)\right]_{g^{[1]}=0,\ldots,g^{[N]}=0,h=1}}{\left[\frac{\delta^{m[1]+\ldots+m[N]}}{\delta z^{[1]}_1 \ldots \delta z^{[1]}_{m[1]} \ldots \delta z^{[N]}_1 \ldots \delta z^{[N]}_{m[N]}} e^{\beta[g^{[1]},\ldots,g^{[N]},h]}\right]_{g^{[1]}=0,\ldots,g^{[N]}=0,h=1}} \tag{41}$$

where:

- *$D_{k+1|k+1}(x) = D_{k+1|k+1}(x|Z^{(k+1)})$ is the PHD of the data updated multi-target RFS $\Xi_{k+1|k+1}$;*
- *β is the multi-sensor cross-term given by definition 5.1.*

The proof is given in 8.15.

Equation (41) can be computed through the *sole* cross-terms defined above. For example, assuming that there are three sensors, that the first one produced a single measurement $z^{[1]}_1$, the second produced two measurements $z^{[2]}_1, z^{[2]}_2$ and the third one none. Then, the data update equation gives:

$$D_{k+1|k+1}(x) = \frac{\left[\frac{\delta}{\delta x}\left(\frac{\delta^3}{\delta z^{[1]}_1 \delta z^{[2]}_1 \delta z^{[2]}_2} e^{\beta[g^{[1]},g^{[2]},g^{[3]},h]}\right)\right]_{g^{[1]}=0,g^{[2]}=0,g^{[3]}=0,h=1}}{\left[\frac{\delta^3}{\delta z^{[1]}_1 \delta z^{[2]}_1 \delta z^{[2]}_2} e^{\beta[g^{[1]},g^{[2]},g^{[3]},h]}\right]_{g^{[1]}=0,g^{[2]}=0,g^{[3]}=0,h=1}} \tag{42}$$

A closer look at the denominator gives:

$$\begin{aligned} & e^{-\beta[0,0,0,1]} \left[\frac{\delta^3}{\delta z^{[1]}_1 \delta z^{[2]}_1 \delta z^{[2]}_2} e^{\beta[g^{[1]},g^{[2]},g^{[3]},h]}\right]_{g^{[1]}=0,g^{[2]}=0,g^{[3]}=0,h=1} \\ &= \beta[\delta_{z^{[1]}_1},0,0,1]\beta[0,\delta_{z^{[2]}_1},0,1]\beta[0,\delta_{z^{[2]}_2},0,1] + \beta[\delta_{z^{[1]}_1},\delta_{z^{[2]}_1},0,1]\beta[0,\delta_{z^{[2]}_2},0,1] + \beta[\delta_{z^{[1]}_1},\delta_{z^{[2]}_2},0,1]\beta[0,\delta_{z^{[2]}_1},0,1] \end{aligned} \tag{43}$$

Note that the denominator is the sum of three *combinational terms*, each one is the product of a various number of cross-terms such that among each combinational term, each measurement appears *once and only once*. Each combinational term represents the likelihood that a particular combination of sources produced the current measurements. For example, $\beta[\delta_{z^{[1]}_1},0,0,1]\beta[0,\delta_{z^{[2]}_1},0,1]\beta[0,\delta_{z^{[2]}_2},0,1]$ is the likelihood that each measurement was produced by a different source (either an underlying target or false alarm process). The second term $\beta[\delta_{z^{[1]}_1},\delta_{z^{[2]}_1},0,1].\beta[0,\delta_{z^{[2]}_2},0,1]$ is the likelihood that a single underlying target produced both measurements $z^{[1]}_1$ and $z^{[2]}_1$, whereas the third measurement was produced by another source (either another target or a false alarm process).

Likewise, the numerator can be expanded as follows:

$$e^{-\beta[0,0,0,1]} \left[\frac{\delta}{\delta x} \left(\frac{\delta^3}{\delta z_1^{[1]} \delta z_1^{[2]} \delta z_2^{[2]}} e^{\beta[g^{[1]},g^{[2]},g^{[3]},h]} \right) \right]_{g^{[1]}=0,g^{[2]}=0,g^{[3]}=0,h=1}$$

$$= \beta[\delta_{z_1^{[1]}}, 0, 0, \delta_x]\beta[0, \delta_{z_1^{[2]}}, 0, 1]\beta[0, \delta_{z_2^{[2]}}, 0, 1] + \beta[\delta_{z_1^{[1]}}, 0, 0, 1]\beta[0, \delta_{z_1^{[2]}}, 0, \delta_x]\beta[0, \delta_{z_2^{[2]}}, 0, 1]$$
$$+ \beta[\delta_{z_1^{[1]}}, 0, 0, 1]\beta[0, \delta_{z_1^{[2]}}, 0, 1]\beta[0, \delta_{z_2^{[2]}}, 0, \delta_x] + \beta[\delta_{z_1^{[1]}}, \delta_{z_1^{[2]}}, 0, \delta_x]\beta[0, \delta_{z_2^{[2]}}, 0, 1]$$
$$+ \beta[\delta_{z_1^{[1]}}, \delta_{z_1^{[2]}}, 0, 1]\beta[0, \delta_{z_2^{[2]}}, 0, \delta_x] + \beta[\delta_{z_1^{[1]}}, \delta_{z_2^{[2]}}, 0, \delta_x]\beta[0, \delta_{z_1^{[2]}}, 0, 1] + \beta[\delta_{z_1^{[1]}}, \delta_{z_2^{[2]}}, 0, 1]\beta[0, \delta_{z_1^{[2]}}, 0, \delta_x] \quad (44)$$

Again, each combinational term represents the likelihood that a particular combination of sources produced the current measurements, such that one of the sources is a target with state x. For example, the term $\beta[\delta_{z_1^{[1]}}, 0, 0, \delta_x]\beta[0, \delta_{z_1^{[2]}}, 0, 1]\beta[0, \delta_{z_2^{[2]}}, 0, 1]$ is the likelihood that a target is in point x and produced measurement $z_1^{[1]}$, whereas each one of the remaining measurements was produced by a different source (either an underlying target or false alarm process).

5.2 Multi-sensor data update equation: combinational expression

Essentialy, as shown in the example above, the set differentiation in (41) produces the cross-terms corresponding to every possible combination regarding the association between current measurements and (eventual) targets and/or false alarms. In the expression of the denominator above (43), a combinational term such as $\beta[\delta_{z_1^{[1]}}, \delta_{z_1^{[2]}}, 0, 1]\beta[0, \delta_{z_2^{[2]}}, 0, 1]$ can be associated to a set of tuples $\{(z_1^{[1]}, z_1^{[2]}), (z_2^{[2]})\}$ where each measurement appear exactly once and where each tuple has at most one measurement per observation space; likewise, each cross-term such as $\beta[\delta_{z_1^{[1]}}, \delta_{z_1^{[2]}}, 0, 1]$ can be associated to a tuple $(z_1^{[1]}, z_1^{[2]})$. In order to get the combinational expression from the multi-sensor data update equation the following notations will be needed. Assuming that, at current time, the sensors produced measurement sets $Z^{[1]}$,...,$Z^{[N]}$:

- $Z_M = \displaystyle\bigcup_{i=1}^{M} Z^{[i]}$ the union of the measurement sets produced by the M first sensors;
- $\check{Z}_M^p = \displaystyle\bigcup_{\substack{I \subseteq [1\ M] \\ |I|=p}} \left(\prod_{i \in I} Z^{[i]} \right)$ the set of all unordered p-tuples defined on the M first measurement sets;
- $\check{Z}_M = \displaystyle\bigcup_{p=1}^{M} \check{Z}_M^p$ the set of all (unordered) tuples defined on the M first measurement sets.

Since, as shown in the example before, each measurement appears once only in each combinational term of in the multi-sensor data update equation, the set of all possible combinational terms $\mathcal{T}(\check{Z}_M)$ must be defined as:

$$\mathcal{T}(\check{Z}_M) = \left\{ T \subseteq \check{Z}_M | \sup_M(T) = \prod_{z \in Z_M} (z, 1) \right\} \tag{45}$$

where $\sup_M(T)$ is the support of $T \subseteq \check{Z}_M$, i.e. the total number of occurences of each measurement in every tuple belonging to T:

$$\sup_M : \quad \mathcal{P}(\check{Z}_M) \to \quad (Z_M \times \mathbb{M})^{|Z_M|} \tag{46}$$

$$T \mapsto \quad \sup_M(T) = \prod_{z \in Z_M} (z, \sum_{T_i \in T} 1_{T_i}(z)) \tag{47}$$

Note that, as explained above, each cross-term $\beta[0, ..., \delta_{z^{[t_1]}}, ..., \delta_{z^{[t_M]}}, ..., h]$ can be represented by the corresponding tuple $(z^{[t_1]}, ..., z^{[t_M]}) \in \check{Z}_N$ [10]. Therefore, cross-terms will be denoted by their equivalent expression:

$\forall\ T = (z^{[t_1]}, ..., z^{[t_M]}) \in \check{Z}_N$

$$\begin{cases} \tilde{\beta}[T, h] = \beta[g^{[1]}, ..., \delta_{z^{[t_1]}}, ..., \delta_{z^{[t_M]}}, ..., g^{[N]}, h] \\ \beta[T, h] = \beta[0, ..., \delta_{z^{[t_1]}}, ..., \delta_{z^{[t_M]}}, ..., 0, h] \end{cases} \tag{48}$$

And, as a convention:

$$\begin{cases} \tilde{\beta}[\emptyset, h] = \beta[g^{[1]}, ..., g^{[N]}, h] \\ \beta[\emptyset, h] = \beta[0, ..., 0, h] \end{cases} \tag{49}$$

The following lemma will be useful in the construction of the multi-sensor data update equation, because it shows that the set of combinational terms can be constructed recursively with the number of sensors M:

Lemma 5.1. *For $M < N$, let $\mathcal{T}(\check{Z}_M)$ be the set of combinational terms considering the M first sensors and let $Z^{[M+1]} = \{z_1^{[M+1]}, ..., z_{m[M+1]}^{[M+1]}\}$ be the measurement set from the next sensor. Then:*

$$\mathcal{T}(\check{Z}_{M+1}) = \bigcup_{T \in \mathcal{T}(\check{Z}_M)} \bigcup_{p=0}^{\min(|T|, m[M+1])} \bigcup_{\substack{I \subseteq [1\ m[M+1]] \\ J \subseteq [1\ |T|] \\ |I|=|J|=p}} \bigcup_{\sigma \in Bij(I,J)} \left(\{T_j\}_{j \notin J} \cup \{(z_i^{[M+1]})\}_{i \notin I} \cup \{(z_i^{[M+1]}, T_{\sigma(i)})\}_{i \in I} \right) \tag{50}$$

where, for any measurement $z^{[M+1]} \in Z^{[M+1]}$ and any (unordered) p-tuple $T = (z^{[t_1]}, ..., z^{[t_p]}) \in \check{Z}_M$, $(z^{[M+1]}, T)$ is the (unordered) $p+1$-tuple $(z^{[t_1]}, ..., z^{[t_p]}, z^{[M+1]}) \in \check{Z}_{M+1}$.

The proof is given in 8.16.

[10] From now on, indexes t_i in tuples are assumed in increasing order by default

Then, the following theorem gives the combinational expression of the data update equation given in proposition (5.2):

Proposition 5.3. *Assuming that the time updated distribution $f_{k+1|k}(X|Z^{(k)})$ is approximately Poisson with parameter μ and intensity $\mu s(x)$, the data update equation is given by:*

$$D_{k+1|k+1}(x) = \beta[\emptyset, \delta_x] + \frac{\displaystyle\sum_{T \in \mathcal{T}(\check{Z}_N)} \sum_{T_i \in T} \left(\beta[T_i, \delta_x] \prod_{\substack{T_j \in T \\ T_j \neq T_i}} \beta[T_j, 1] \right)}{\displaystyle\sum_{T \in \mathcal{T}(\check{Z}_N)} \prod_{T_i \in T} \beta[T_i, 1]} \tag{51}$$

where:

- $D_{k+1|k+1}(x) = D_{k+1|k+1}(x|Z^{(k+1)})$ *is the PHD of the data updated multi-target RFS* $\Xi_{k+1|k+1}$*;*
- β *is the multi-sensor cross-term given by definition 5.1;*
- $\mathcal{T}(\check{Z}_N)$ *is the set of combinational terms given by* (45).

The proof is given in 8.17.

6 Simplification of the multi-sensor data update equation

Clearly the computation becomes tedious when the number of sensor and/or measurements per sensor is large enough, because the number of cross-terms, the number and size of combinational terms enlarge too. The following sections deals with some leads to reduce the computational cost of the data update step.

6.1 Simplification by target space partitioning

Although the data update equations (41) and (51) involves many cross-terms as seen in examples (43) and (44), some of them may equal to zero. Finding such cross-terms is interesting because it reduces the number of combinational term to consider (since whenever a cross-term vanishes, every combinational term it belongs to vanishes as well) and hence simplifies the computation of the data update equation. Moreover, the simplified expression provides the *exact same result* as the data update equation, since only void combinational terms are discarded.

This section deals with a solution which discards void cross-terms based on the particular configuration of the sensors at the current time. In the common case where sensors' field of views (FOVS) are bounded in state space and may not completely overlap each other, one can have the intuitive feeling that the PHD $D_{k+1|k+1}$ can be updated in any subregion $S \subseteq \mathcal{X}$ without taking into account sensors whose FOVs are "away" from S. The idea is to set a partition of both the sensors and the state space, such that each element in the sensor partition is linked to an element in the state space partition. Then, the PHD $D_{k+1|k+1}$ computed in x can be reduced by considering the points form the same partition element than x and the sensors from the corresponding element in the sensor partition.

Definition 6.1. *For any sensor $j \in [1\ N]$, let $F^{[j]}_{k+1} \subseteq \mathcal{X}$ denote its field of view at time $k+1$ defined as:*

$$\forall x \in \mathcal{X}, x \in F^{[j]}_{k+1} \Leftrightarrow p^{[j]}_{d,k+1}(x) \neq 0 \tag{52}$$

Define the equivalence relation "cross" ($\leftrightarrow$) between sensors as:

$$\forall i, j \in [1\ N], (i \leftrightarrow j) \Leftrightarrow (F^{[i]}_{k+1} \cap F^{[j]}_{k+1} \neq \emptyset) \tag{53}$$

Let $\{P_s(p)\}_{p=1}^{P}$ be the sensor partition of $[1\ N]$ formed by the equivalence classes of the transitive closure of the "cross" relation. Let $\{P_t(p)\}_{p=0}^{P}$ be the space partition of the state space $\mathcal{X}$ defined by: [11]

$$P_t(p) = \begin{cases} \overline{\bigcup_{j=1}^{N} F^{[j]}_{k+1}} & (p = 0) \\ \bigcup_{j \in P_s(p)} F^{[j]}_{k+1} & (p \neq 0) \end{cases} \tag{54}$$

Finally, for any element $P_s(p)$ of the sensor partition, let $N_p = |P_s(p)|$ denotes the number of sensors in $P_s(p)$, and let $p_1, ..., p_{N_p}$ denote the increasing indexes in $[1\ N]$ of sensors belonging to $P_s(p)$.

[11] $\{P_s(p)\}_{p=1}^{P} = \{P_{S,k+1}(p)\}_{p=1}^{P_{k+1}}$ and $\{P_t(p)\}_{p=0}^{P} = \{P_{T,k+1}(p)\}_{p=0}^{P_{k+1}}$, time subscripts are omitted for simplicity's sake.

In the defintion above, it is assumed that $\forall j \in [1\ N]$, $F_{k+1}^{[j]} \neq \emptyset$. Should any sensor have a void field of view at the current time, it would be ignored for the data update step, discarded from the sensor indexes $[1\ N]$ which would be reindexed accordingly. Note also that if the combined FOV of all the sensors is the whole state space, $P_t(0)$ is empty and should not be considered as an element of the state space partition.

The following definition introduces the "bounded" cross-terms whose "scope" are limited to one element of the sensor partition $P_s(p)$ and to the corresponding element in the state space partition $P_t(p)$.

Definition 6.2. *For any $p \in [1\ P]$, define the generic cross-term bounded to element $P_s(p)$ as:*

$$\beta_p[g^{[p_1]}, ..., g^{[p_{N_p}]}, h] = \sum_{j \in P_s(p)} (\lambda^{[j]} c^{[j]}[g^{[j]}] - \lambda^{[j]}) + \mu s \left[h \prod_{j \in P_s(p)} (1 - p_d^{[j]} + p_d^{[j]} p_{g^{[j]}}^{O,[j]}) \right]_p - \mu$$

where $s[h]_p$ is the restriction of $s[h]$ to $P_t(p) \subseteq \mathcal{X}$, i.e. $s[h]_p = s[1_{P_t(p)} h] = \int_{P_t(p)} h(x) s(x) dx$.
Let $x \in \mathcal{X}$ be any element in target space. The bounded cross-term set in target state x is defined by:

$$\beta_p[g^{[p_1]}, ..., g^{[p_{N_p}]}, \delta_x] = \frac{\delta}{\delta x} \beta_p[g^{[p_1]}, ..., g^{[p_{N_p}]}, h]$$

Let $1 \leqslant M \leqslant N$ and $t_1, ...t_M$ be the increasing indexes in $[1\ N]$ of any M distinct sensors. For any sensor t_j, let $z^{[t_j]} \in Z_{k+1}^{[t_j]}$ be one of the current measurements of sensor t_j, where $Z_{k+1}^{[t_j]}$ is assumed non empty. The bounded cross-term set in $z^{[t_1]}, ..., z^{[t_M]}$ is defined by:

$$\beta_p[g^{[p_1]}, ..., \delta_{z^{[t_1]}}, ..., \delta_{z^{[t_M]}}, ..., g^{[p_{N_p}]}, h] = \frac{\delta^M}{\delta z^{[t_1]} ... \delta z^{[t_M]}} \beta_p[g^{[p_1]}, ..., g^{[p_{N_p}]}, h]$$

The bounded cross-term set in measurements $z^{[t_1]}, ..., z^{[t_M]}$ and target state x is defined by:

$$\begin{aligned}\beta_p[g^{[p_1]}, ..., \delta_{z^{[t_1]}}, ..., \delta_{z^{[t_M]}}, ..., g^{[p_{N_p}]}, \delta_x] &= \frac{\delta^M}{\delta z^{[t_1]} ... \delta z^{[t_M]}} \beta_p[g^{[p_1]}, ..., g^{[p_{N_p}]}, \delta_x] \\ &\left(= \frac{\delta}{\delta x} \beta[g^{[p_1]}, ..., \delta_{z^{[t_1]}}, ..., \delta_{z^{[t_M]}}, ..., g^{[p_{N_p}]}, h]\right)\end{aligned}$$

That is, the definition of the bounded cross-terms is identical to the definitions 5.1 and 5.2 of multi-sensor cross-terms except that only the sensors belonging to an element $P_s(p)$ are considered. Likewise, proposition 5.1 applies to bounded cross terms, where all but the sensors in $P_s(p)$ are ignored and the "expectation" is restricted to the subset $P_t(p) \subseteq \mathcal{X}$.

The following proposition is fundamental because it provides tools for reducing the general multi-sensor cross-terms to the bounded ones:

Proposition 6.1. *With the same notations as in definitions 5.2 and 6.2, we have:*

$$\beta[g^{[1]}, ..., g^{[N]}, \delta_x] = \begin{cases} \beta_p[g^{[p_1]}, ..., g^{[p_{N_p}]}, \delta_x] & (\exists p \in [1\ P], x \in P_t(p)) \\ \mu s(x) & (x \in P_t(0)) \end{cases}$$

$$\beta[g^{[1]}, ..., \delta_{z^{[t_1]}}, ..., \delta_{z^{[t_M]}}, ..., g^{[N]}, h] = \begin{cases} \beta_p[g^{[p_1]}, ..., \delta_{z^{[t_1]}}, ..., \delta_{z^{[t_M]}}, ..., g^{[p_{N_p}]}, h] & (\exists p \in [1\ P], \{t_1, ..., t_M\} \subseteq P_s(p)) \\ 0 & (otherwise) \end{cases}$$

$$\beta[g^{[1]}, ..., \delta_{z^{[t_1]}}, ..., \delta_{z^{[t_M]}}, ..., g^{[N]}, \delta_x] = \begin{cases} \beta_p[g^{[p_1]}, ..., \delta_{z^{[t_1]}}, ..., \delta_{z^{[t_M]}}, ..., g^{[p_{N_p}]}, \delta_x] & (\exists p \in [1\ P], \{t_1, ..., t_M\} \subseteq P_s(p), x \in P_t(p)) \\ 0 & (otherwise) \end{cases}$$

The proof is given in section 8.18.

The results above have an intuitive interpretation once the fuzzy functionals $g^{[j]}$ are set to zero and the fuzzy functional h set to one. First of all, recall that $\beta[0, ..., 0, \delta_x]$ can be seen as the likelihood that there is a target in point x *and* that this target was undetected by all the sensors. If this target was outside the combined FOV of all the sensors (i.e. $x \in P_t(0)$) then the fact that the target remained undetected brings no information as this event was to occur with probability one. Therefore,$\beta[0, ..., 0, \delta_x]$ is exactly the intensity of the PHD $D_{k+1|k}$ before the data update, i.e. $\mu s(x)$. On the other hand, if a target x was inside the combined FOV of a sensor partition (i.e. $x \in P_t(p)$), then the likelihood that this target remained undetected by *all* the sensors is exactly the likelihood that this target remained undetected by the sensors of $P_s(p)$ *only*, since x was to be undetected by sensors of other partition elements with probability one. Therefore, $\left[\beta[g^{[1]}, ..., g^{[N]}, \delta_x]\right]_{g^{[1]}=0,...,g^{[N]}=0} = \left[\beta_p[g^{[p_1]}, ..., g^{[p_{N_p}]}, \delta_x]\right]_{g^{[p_1]}=0,...,g^{[p_{N_p}]}=0}$.

Likewise, the reformulation of $\beta[0, ..., \delta_{z^{[t_1]}}, ..., \delta_{z^{[t_M]}}, ..., 0, \delta_x]$ has a straightforward interpretation. Recall that it can be seen as the likelihood that a target is in point x, that it was detected by sensors $t_1, ..., t_M$ only *and* that it produced measurements $z^{[t_1]}, ..., z^{[t_M]}$. Since a target inside subset $P_t(p)$ can be detected by no sensors but the ones inside subset $P_s(p)$, it is clear that $\beta[0, ..., \delta_{z^{[t_1]}}, ..., \delta_{z^{[t_M]}}, ..., 0, h] = 0$ whenever sensors $t_1, ..., t_M$ are not from the same subset.

If they all belong to the same subset $P_s(p)$ *but* the target belongs to a subset $P_t(q)$, where $q \neq p$, then target x is undetected by these sensors with probability one, and therefore the likelihood vanishes as well. However, if all the sensors $t_1, ..., t_M$ belong to the same subset $P_s(p)$ *and* the target belongs to the corresponding subset $P_t(p)$, then the likelihood $\beta[0, ..., \delta_{z^{[t_1]}}, ..., \delta_{z^{[t_M]}}, ..., 0, \delta_x]$ equals to the likelihood limited to subsets $P_s(p)$ and $P_s(p)$, i.e., $\beta_p[0, ..., \delta_{z^{[t_1]}}, ..., \delta_{z^{[t_M]}}, ..., 0, \delta_x]$. The same arguments are relevant for the reformulation of the expectation $\beta[0, ..., \delta_{z^{[t_1]}}, ..., \delta_{z^{[t_M]}}, ..., 0, 1]$.

Likewise, the tuples and combinational terms introduced in section 5.2 have "bounded" equivalents. Assuming that, at current time, the sensors belonging to partition element $P_s(p)$ produced measurement sets $Z^{[p_1]}$,...,$Z^{[p_{N_p}]}$:

- $Z_M^{(p)} = \bigcup_{i=1}^{M} Z^{[p_i]}$ is the union of the measurement sets produced by the M first sensors in $P_s(p)$;
- $\check{Z}_M^{(p),q} = \bigcup_{\substack{I \subseteq [1\ M] \\ |I|=q}} \left(\prod_{i \in I} Z^{[p_i]} \right)$ the set of all unordered q-tuples defined on the M first measurement sets;
- $\check{Z}_M^{(p)} = \bigcup_{q=1}^{M} \check{Z}_M^{(p),q}$ the set of all (unordered) tuples defined on the M first measurement sets.

As in equation (45), the set of all possible bounded combinational terms $\mathcal{T}(\check{Z}_{N_p}^{(p)})$ is defined as:

$$\mathcal{T}(\check{Z}_{N_p}^{(p)}) = \left\{ T \subseteq \check{Z}_{N_p}^{(p)} | \sup{}_{N_p}^{(p)}(T) = \prod_{z \in Z_{N_p}^{(p)}} (z, 1) \right\} \tag{55}$$

where $\sup_{N_p}^{(p)}(T)$ is the support of $T \subseteq \check{Z}_{N_p}^{(p)}$, i.e. the total number of occurences of each measurement in every tuple belonging to T:

$$\sup{}_{N_p}^{(p)} : \qquad \mathcal{P}(\check{Z}_{N_p}^{(p)}) \to \qquad (Z_{N_p}^{(p)} \times \mathbb{N})^{|Z_{N_p}^{(p)}|} \tag{56}$$

$$T \mapsto \qquad \sup{}_{N_p}^{(p)} = \prod_{z \in Z_{N_p}^{(p)}} (z, \sum_{T_i \in T} 1_{T_i}(z)) \tag{57}$$

Likewise, each bounded cross-term $\beta_p[0, ..., \delta_{z^{[t_1]}}, ..., \delta_{z^{[t_M]}}, ..., h]$ can be represented by the corresponding unordered tuple $(z^{[t_1]}, ..., z^{[t_M]}) \in \check{Z}_{N_p}^{(p)}$. Therefore, bounded cross-terms will be denoted by their equivalent expression:

$\forall p \in [1\ P], \forall\ T = (z^{[t_1]}, ..., z^{[t_M]}) \in \check{Z}_{N_p}^{(p)}$

$$\begin{cases} \tilde{\beta}_p[T, h] = \beta_p[g^{[p_1]}, ..., \delta_{z^{[t_1]}}, ..., \delta_{z^{[t_M]}}, ..., g^{[p_{N_p}]}, h] \\ \beta_p[T, h] = \beta_p[0, ..., \delta_{z^{[t_1]}}, ..., \delta_{z^{[t_M]}}, ..., 0, h] \end{cases} \tag{58}$$

And, as a convention, $\forall p \in [1\ P]$:

$$\begin{cases} \tilde{\beta}_p[\emptyset, h] = \beta_p[g^{[p_1]}, ..., g^{[p_{N_p}]}, h] \\ \beta_p[\emptyset, h] = \beta_p[0, ..., 0, h] \end{cases} \tag{59}$$

Lemma 5.1 still holds for each partition element $P_s(p)$; that is, $\mathcal{T}(\check{Z}_{N_p}^{(p)})$ can be constructed recursively by considering the measurement sets of the sensors belonging to $P_s(p)$ *only*. Considering the results of proposition 6.1, the data update equation can then be simplified:

Proposition 6.2. *Denote by $\{P_s(p)\}_{p=1}^{P}$ the sensor partition and by $\{P_t(p)\}_{p=0}^{P}$ the corresponding partition as described in definition 6.1. Then proposition 5.3 simplifies as following:*

$$D_{k+1|k+1}(x) = \begin{cases} D_{k+1|k}(x) & (x \in P_t(0)) \\ \beta_p[\emptyset, \delta_x] + \dfrac{\displaystyle\sum_{T \in \mathcal{T}(\check{Z}^{(p)}_{N_p})} \sum_{T_i \in T} \left(\beta_p[T_i, \delta_x] \prod_{\substack{T_j \in T \\ T_j \neq T_i}} \beta_p[T_j, 1] \right)}{\displaystyle\sum_{T \in \mathcal{T}(\check{Z}^{(p)}_{N_p})} \prod_{T_i \in T} \beta_p[T_i, 1]} & (x \in P_t(p), p \neq 0) \end{cases} \tag{60}$$

where:

- $D_{k+1|k+1}(x) = D_{k+1|k+1}(x|Z^{(k+1)})$ *is the PHD of the Bayes updated multitarget RFS* $\Xi_{k+1|k+1}$*;*
- $D_{k+1|k}(x) = D_{k+1|k}(x|Z^{(k+1)})$ *is the PHD of the time updated multitarget RFS* $\Xi_{k+1|k}$*;*
- $\forall p \in [0\ P]$*,* β_p *is the bounded cross-term given by definition 6.2;*
- $\forall p \in [1\ P]$*,* $\mathcal{T}(\check{Z}^{(p)}_{N_p})$ *is the set of combinational terms given by* (55)*.*

The proof is given in section 8.19.

In its set derivative form, proposition 6.2 gives:

Corollary 6.1. *Denote by $\{P_s(p)\}_{p=1}^{P}$ the sensor partition and by $\{P_t(p)\}_{p=0}^{P}$ the corresponding partition as described in definition 6.1 and by $Z^{[j]}_{k+1} = \{z^{[j]}_1, ..., z^{[j]}_{m[j]}\}$ the set of current measurements produced by each sensor j. Then proposition 5.2 simplifies as following:*

$$D_{k+1|k+1}(x) = \begin{cases} D_{k+1|k}(x) & (x \in P_t(0)) \\ \dfrac{\left[\dfrac{\delta}{\delta x} \left(\dfrac{\delta^{m[p_1]+\ldots+m[p_{N_p}]}}{\delta z^{[p_1]}_1 \ldots \delta z^{[p_1]}_{m[p_1]} \ldots \delta z^{[p_{N_p}]}_1 \ldots \delta z^{[p_{N_p}]}_{m[p_{N_p}]}} e^{\beta_p[g^{[p_1]}, \ldots, g^{[p_{N_p}]}, h]} \right) \right]_{g^{[p_1]}=0, \ldots, g^{[p_{N_p}]}=0, h=1}}{\left[\dfrac{\delta^{m[p_1]+\ldots+m[p_{N_p}]}}{\delta z^{[p_1]}_1 \ldots \delta z^{[p_1]}_{m[p_1]} \ldots \delta z^{[p_{N_p}]}_1 \ldots \delta z^{[p_{N_p}]}_{m[p_{N_p}]}} e^{\beta_p[g^{[p_1]}, \ldots, g^{[p_{N_p}]}, h]} \right]_{g^{[p_1]}=0, \ldots, g^{[p_{N_p}]}=0, h=1}} & (x \in P_t(p), p \neq 0) \end{cases} \tag{61}$$

where:

- $D_{k+1|k+1}(x) = D_{k+1|k+1}(x|Z^{(k+1)})$ *is the PHD of the Bayes updated multitarget RFS* $\Xi_{k+1|k+1}$*;*
- $D_{k+1|k}(x) = D_{k+1|k}(x|Z^{(k+1)})$ *is the PHD of the time updated multitarget RFS* $\Xi_{k+1|k}$*;*
- β_p *is the bounded cross-term given by definition 6.2.*

The proof follows immediately from proposition 6.2.

The gain in computational cost when using the partitioning method (see (61) or (60)) rather than the general equation ((41) or (51)) seems quite difficult to evaluate, because we lack a clear expression of number of terms and cross-terms we need to compute. Recall that updating the PHD using (41) or (61) requires the generation of terms which are the products of general multi-sensor cross-terms (for (41)) or bounded ones (for (61)). To be sure, the number of different cross-terms will be reduced with the partition method, but the significant computational gain is likely to stem from the tedious generation of the combinational terms composing the numerator (see example (44)) and the denominator (see example (43)) of the data update equation. Still, some elements of comparison can be given considering:

- The number of *different* cross-terms computed in each case;
- The computational cost of the partition step.

Assume that, at current step, each sensor i produces an average of M measurements. Assume also that the PHD is implemented with a particle filter, i.e. the time updated density $D_{k+1|k}$ is approximated by a collection of weighted particles $\{w_j, x_j\}$, the current number of particles being K.

Without the partition step, the generic cross-term has N observation functionals $g^{[i]}$ and one state functional h. Since each functional $g^{[i]}$ can be differentiated in any current measurement produced by sensor i, the number of cross-terms set in a given particle x_j is roughly M^N. Since the cross-terms which denote "expectations" over state space are approximated by the sum of the corresponding cross-terms set on each particle (e.g. $\beta[\delta_{z_1^{[1]}}, \delta_{z_1^{[2]}}, 0, 1] \simeq \sum_{j=1}^{K} \beta[\delta_{z_1^{[1]}}, \delta_{z_1^{[2]}}, 0, \delta_{x_j}]$), the cross-terms set on each particle must be computed; therefore the number of cross-terms N_{np} is given by:

$$N_\beta^{true} \sim \mathcal{O}\left(K.M^N\right) \tag{62}$$

With the partition method, the result is similar except that the overall update is splitted on several "smaller" updates. Assume that partitioning in current step gives P elements, each sensor element has N_p sensors such that $\sum_{p=1}^{P} N_p = N$, and the corresponding particle element has K_p particles such that $\sum_{p=1}^{P} K_p \leqslant K$ (some particles may be outside the combined FOV of all the sensors, i.e. belong to $P_t(0)$). The number of bounded cross-terms to compute is therefore:

$$N_\beta^{part} \sim \mathcal{O}\left(\sum_{p=1}^{P} K_p.M^{N_p}\right) \tag{63}$$

In the best-case scenario where the FOVs are disjoints, we have $P = N$ and therefore $N_p = 1$. The multi-sensor update is merely the union of N independant single-sensor updates and $N_\beta^{part} \sim \mathcal{O}(K.M)$. In the worst-case scenario, all the FOVs are in the same equivalence class and $N_\beta^{part} \sim N_\beta^{true}$. The partition therefore spares the computation of a significant number of cross-terms in general. Keep in mind, though, that the number of cross-terms does not fully reflect the complexity of the data update; the same approach on the number of combinational terms would be quite useful too.

The partition method nonetheless requires the computation of sensor and space partitions *at every step* k. The following paragraph provides the pseudo-code for the partitioning assuming that the PHD is implemented with a particle filter. Since there is a mapping from the sensor partition elements to the state partition elements, and since the number of sensors N is much smaller than the particle number K, it is somewhat easier to compute the sensor partition and deduce the particle partition. The input is a N-by-M binary matrix, called the cover matrix C, where $C(i,j) = 0$ if $p_D^{[i]}(x_j) = 0$ (i.e. particle x_j lies outside the FOV of sensor i) and $C(i,j) = 1$ otherwise. Note that the cover matrix is necessary for the computation of the cross-terms whether the partition method is used or not, thus it should not add up to the computational cost of the partitioning step.

1. Initialize adjacency matrix $A \leftarrow (C.C^T \neq 0)$ [a]
2. Initialize temp matrix $T \leftarrow (A^2 \neq 0)$
3. While $T \neq A$:
    3.1. $A \leftarrow T$
    3.2. $T \leftarrow (A^2 \neq 0)$
4. Initialize partition number $p \leftarrow 0$
5. For $i = 1$ to sensor number N:
    5.1. If $A(i, :) \neq 0$
        5.1.1. If $p = 0$
            5.1.1.1. Increment partition number $p \leftarrow 1$
            5.1.1.2. Initialize sensor partition matrix $S_p(1, :) \leftarrow A(i, :)$
        5.1.2. Elseif $S(:, i) = 0$
            5.1.2.1. Increment partition number $p \leftarrow p + 1$
            5.1.2.2. Add a new partition $S_p(p, :) \leftarrow A(i, :)$
6. Compute particle partition matrix $P_p \leftarrow (S_p.C \neq 0)$

[a] If $A \leftarrow (B \neq 0)$, A is the binary matrix with the same size than B such that $A(i, j) = 1$ if $B(i, j) \neq 0$, $A(i, j) = 0$ otherwise.

When step 1 is reached in the pseudo-code above, the N-by-N adjacency matrix A is a matrix representation of the cross relation as defined in (53), i.e. $A(i, j) = 1$ if and only if sensor i and j and overlapping FOVs, $A(i, j) = 0$ otherwise. Note that, since the state space is discretised by the collection of particles, two FOVs overlap if and only if there is at least one common particle x belonging to both FOVs. When step 4 is reached, A is the matrix representation of the transitive closure of the cross relation. Hence, $A(i, j) = 1$ if and only if sensor i and j are in the same equivalence class of the transitive closure, i.e, they are in the same sensor partition element. Step 5 simply extracts lines from matrix A to form the sensor partition matrix S_p, which is a P-by-N binary matrix, where P is the number of partition, such that $S_p(i, j) = 1$ if and only if sensor j belongs to partition element i. Finally, step 6 computes the particle partition matrix P_p, which is a P-by-K binary matrix such that $P_p(i, j) = 1$ if and only if particle j is inside the combined FOV of the sensor partition element i. Note that some particle x_j may be outside the combined FOV of all the sensors (i.e. $x_j \in P_t(0)$), in such a case column $P_p(:, j)$ is null in the partition matrix. Although it is far less likely, a sensor may fail to appear in any sensor partition element if its FOV contains no particle at all; such a sensor will be naturally discarded in the data update equation.

Note that the partition step has a relatively low computational cost because the bulk of the processing involves square matrices whose size does not excess the number of sensors N. To be sure, the following step is the PHD data update for each particle which requires the computation of cross-terms bounded to the current particle partition element and the corresponding sensor partition element; this require many memory accesses to the sensor and particle partition matrices which should add up to the overall computational cost of the partition method. A thorough comparison of the complexity of the PHD data update with and without the partition step would require at least Monte Carlo runs of both methods; a formal expression of the complexity of both PHD data update equations would be valuable too.

6.2 Approximation by restricting hypotheses

Assume that there are $N = 3$ sensors, that the first sensor produced one measurement $z_1^{[1]}$, the second sensor one measurement $z_1^{[2]}$ and the last sensor two measurements $z_1^{[3]}, z_2^{[3]}$. Then, as in example (43), the denominator of the PHD data update can be expanded as follows:

$$
\begin{aligned}
& e^{-\beta[0,0,0,1]} \left[\frac{\delta^4}{\delta z_1^{[1]} \delta z_1^{[2]} \delta z_1^{[3]} \delta z_2^{[3]}} e^{\beta[g^{[1]},g^{[2]},g^{[3]},h]} \right]_{g^{[1]}=0,g^{[2]}=0,g^{[3]}=0,h=1} = \\
1^{st} - order\ term & \left\{ \beta[\delta_{z_1^{[1]}},0,0,1] \,.\, \beta[0,\delta_{z_1^{[2]}},0,1] \,.\, \beta[0,0,\delta_{z_1^{[3]}},1] \,.\, \beta[0,0,\delta_{z_2^{[3]}},1] \right. \\
2^{nd} - order\ terms & \left\{ \begin{array}{l}
+ \beta[\delta_{z_1^{[1]}},0,0,1] \,.\, \beta[0,\delta_{z_1^{[2]}},\delta_{z_2^{[3]}},1] \,.\, \beta[0,0,\delta_{z_1^{[3]}},1] \\
+ \beta[\delta_{z_1^{[1]}},0,\delta_{z_2^{[3]}},1] \,.\, \beta[0,\delta_{z_1^{[2]}},0,1] \,.\, \beta[0,0,\delta_{z_1^{[3]}},1] \\
+ \beta[\delta_{z_1^{[1]}},\delta_{z_1^{[2]}},0,1] \,.\, \beta[0,0,\delta_{z_1^{[3]}},1] \,.\, \beta[0,0,\delta_{z_2^{[3]}},1] \\
+ \beta[0,\delta_{z_1^{[2]}},\delta_{z_1^{[3]}},1] \,.\, \beta[\delta_{z_1^{[1]}},0,0,1] \,.\, \beta[0,0,\delta_{z_2^{[3]}},1] \\
+ \beta[\delta_{z_1^{[1]}},0,\delta_{z_1^{[3]}},1] \,.\, \beta[0,\delta_{z_1^{[2]}},0,1] \,.\, \beta[0,0,\delta_{z_2^{[3]}},1] \\
+ \beta[0,\delta_{z_1^{[2]}},\delta_{z_1^{[3]}},1] \,.\, \beta[\delta_{z_1^{[1]}},0,\delta_{z_2^{[3]}},1] \\
+ \beta[\delta_{z_1^{[1]}},0,\delta_{z_1^{[3]}},1] \,.\, \beta[0,\delta_{z_1^{[2]}},\delta_{z_2^{[3]}},1]
\end{array} \right. \\
3^{rd} - order\ terms & \left\{ \begin{array}{l}
+ \beta[\delta_{z_1^{[1]}},\delta_{z_1^{[2]}},\delta_{z_2^{[3]}},1] \,.\, \beta[0,0,\delta_{z_1^{[3]}},1] \\
+ \beta[\delta_{z_1^{[1]}},\delta_{z_1^{[2]}},\delta_{z_1^{[3]}},1] \,.\, \beta[0,0,\delta_{z_2^{[3]}},1]
\end{array} \right.
\end{aligned} \tag{64}
$$

Since $N = 3$ and each sensor produced at least one measurement, the generic cross-term can be set in one, two, or three measurements. In the expression above, n-th order comnbinational terms are products of cross-terms set in *at most* n measurements, such that at least one cross-term in the product is set in n measurements.

An straightforward approximation of the expression above would be to skip some terms to make the computation of the expression easier. Recall that $\beta[0,...,\delta_{z^{[t_1]}},...,\delta_{z^{[t_M]}},...,0,\delta_x]$ can be seen as the likelihood that a target is in point x, that it was detected by sensors $t_1,...,t_M$ only *and* that it produced measurements $z^{[t_1]},...,z^{[t_M]}$. Therefore, a hypothesis restricting the maximum number of sensors which can detect a single sensor to a given number $D_{max} \leqslant N$ implies that all cross-terms set in more than D_{max} measurements vanish and, therefore, implies that all n-th order terms with $n > D_{max}$ vanish as well. However, restricting the number of detections per target contradicts the independence of sensors' observation processes.

Note that the more drastic approximation, which skip all terms but the first-order one, has the following closed-form expression:

$$\frac{\left[\frac{\delta}{\delta x}\left(\frac{\delta^{m[1]+\ldots+m[N]}}{\delta z_1^{[1]}\ldots\delta z_{m[1]}^{[1]}\ldots\delta z_1^{[N]}\ldots\delta z_{m[N]}^{[N]}}e^{\beta[g^{[1]},\ldots,g^{[N]},h]}\right)\right]_{g^{[1]}=0,\ldots,g^{[N]}=0,h=1}}{\left[\frac{\delta^{m[1]+\ldots+m[N]}}{\delta z_1^{[1]}\ldots\delta z_{m[1]}^{[1]}\ldots\delta z_1^{[N]}\ldots\delta z_{m[N]}^{[N]}}e^{\beta[g^{[1]},\ldots,g^{[N]},h]}\right]_{g^{[1]}=0,\ldots,g^{[N]}=0,h=1}}$$

$$= \frac{\left[\frac{\delta}{\delta x}\left(e^{\beta[g^{[1]},\ldots,g^{[N]},h]}\prod_{j=1}^{N}\prod_{i=1}^{m[j]}\left(\beta[g^{[1]},...,g^{[j-1]},\delta_{z_i^{[j]}},g^{[j+1]},...,g^{[N]},h]\right)\right)\right]_{g^{[1]}=0,\ldots,g^{[N]}=0,h=1}}{\left[e^{\beta[g^{[1]},\ldots,g^{[N]},h]}\prod_{j=1}^{N}\prod_{i=1}^{m[j]}\left(\beta[g^{[1]},...,g^{[j-1]},\delta_{z_i^{[j]}},g^{[j+1]},...,g^{[N]},h]\right)\right]_{g^{[1]}=0,\ldots,g^{[N]}=0,h=1}}$$

$$= \frac{\beta[0,...,0,\delta_x]e^{\beta[0,\ldots,0,1]}\prod_{j=1}^{N}\prod_{i=1}^{m[j]}\left(\beta[0,...,0,\delta_{z_i^{[j]}},0,...,0,1]\right)}{e^{\beta[0,\ldots,0,1]}\prod_{j=1}^{N}\prod_{i=1}^{m[j]}\left(\beta[0,...,0,\delta_{z_i^{[j]}},0,...,0,1]\right)}$$

$$+ \frac{e^{\beta[0,\ldots,0,1]}\prod_{j=1}^{N}\prod_{i=1}^{m[j]}\left(\beta[0,...,0,\delta_{z_i^{[j]}},0,...,0,1]\right).\sum_{j=1}^{N}\sum_{i=1}^{m[j]}\left(\frac{\beta[0,\ldots,0,\delta_{z_i^{[j]}},0,\ldots,0,\delta_x]}{\beta[0,\ldots,0,\delta_{z_i^{[j]}},0,\ldots,0,1]}\right)}{e^{\beta[0,\ldots,0,1]}\prod_{j=1}^{N}\prod_{i=1}^{m[j]}\left(\beta[0,...,0,\delta_{z_i^{[j]}},0,...,0,1]\right)}$$

$$= \beta[0,...,0,\delta_x] + \sum_{j=1}^{N}\sum_{i=1}^{m[j]}\frac{\beta[0,...,0,\delta_{z_i^{[j]}},0,...,0,\delta_x]}{\beta[0,...,0,\delta_{z_i^{[j]}},0,...,0,1]}$$

$$= \mu\prod_{j=1}^{N}\left(1-p_d^{[j]}(x)\right)s(x) + \sum_{j=1}^{N}\sum_{z\in Z_{k+1}^{[j]}}\frac{\mu\prod_{l\neq j}\left(1-p_d^{[l]}(x)\right)p_d^{[j]}(x)L_z^{[j]}(x)s(x)}{\lambda^{[j]}c^{[j]}(z)+\mu s\left[\prod_{l\neq j}\left(1-p_d^{[l]}\right)p_d^{[j]}L_z^{[j]}\right]}$$

That is:

$$D_{k+1|k+1}(x) \simeq \left(\prod_{j=1}^{N}\left(1-p_d^{[j]}(x)\right) + \sum_{j=1}^{N}\sum_{z\in Z_{k+1}^{[j]}}\frac{\prod_{l\neq j}\left(1-p_d^{[l]}(x)\right)p_d^{[j]}(x)L_z^{[j]}(x)}{\lambda^{[j]}c^{[j]}(z)+\mu s\left[\prod_{l\neq j}\left(1-p_d^{[l]}\right)p_d^{[j]}L_z^{[j]}\right]}\right)D_{k+1|k}(x)$$

Note that this equation is somewhat similar to the single-sensor PHD data update (36). Indeed, since no target may be detected by more than one sensor, each target produces at most one measurement in Z_{k+1}, the union of the measurement sets from each sensor. Therefore, the combination of all the sensors may be seen as a "pseudo-sensor"; however, the quantities $\prod_{l\neq j}\left(1-p_d^{[l]}(x)\right).p_d^{[j]}(x)$ are unconsistent because they cannot be interpreted as detection probabilities. Mahler proposed in [2] another approximation based on a "pseudo-sensor" which encapsulates all the sensors in a single observation process, but it also contradicts the single observation processes since any target produces at most one measurement when detected.

6.3 Approximation by sequential filtering

Since the complexity in using the general data update formula (41) seems to arise from the fact that the cross-terms involve the measurements from *all* the sensors *in a whole*, several approximations can be proposed by processing the sensors *sequentially*. In this section, three such approximations are presented.

6.3.1 Product approximation

The simplest approximation is the *product approximation*, it seems to be the solution proposed by Mahler in [1] :

Definition 6.3. *Assuming that time updated density is Poisson with PHD $D_{k+1|k}(x) = \mu s(x)$, the product approximation of the data update formula is given by:*

$$D_{k+1|k+1}(x) \simeq \left(\prod_{i=1}^{N} C_P^{[i]}(Z_{k+1}^{[i]}|x)\right) D_{k+1|k}(x) \tag{65}$$

where $C_P^{[i]}(Z_{k+1}^{[i]}|x) = 1 - p_d^{[i]} + \sum_{z \in Z_{k+1}^{[i]}} \frac{p_d^{[i]}(x) L_z^{[i]}(x)}{\lambda^{[i]} c^{[i]}(z) + \mu s[p_d^{[i]} L_z^{[i]}]}$ *is the product corrector of the i-th sensor.*

Note that the expression of the product corrector is similar to the single-sensor PHD data update equation (36). Indeed, $C_P^{[i]}$ is computed *independently* of the other correctors, ignoring all the measurements but those produced by sensor i, and we have:

$$C_P^{[i]}(Z_{k+1}^{[i]}|x)\mu s(x) = \frac{\left[\frac{\delta}{\delta x}\left(\frac{\delta^{m[i]}}{\delta z_1^{[i]} \ldots \delta z_{m[i]}^{[i]}} e^{\beta_P^{[i]}[g^{[i]},h]}\right)\right]_{g^{[i]}=0,h=1}}{\left[\frac{\delta^{m[i]}}{\delta z_1^{[i]} \ldots \delta z_{m[i]}^{[i]}} e^{\beta_P^{[i]}[g^{[i]},h]}\right]_{g^{[i]}=0,h=1}} \tag{66}$$

where $\beta_P^{[i]}[g^{[i]}, h] = \lambda^{[i]} c^{[i]}[g^{[i]}] - \lambda^{[i]} + \mu s[h(1 - p_d^{[i]} + p_d^{[i]}.p_{g^{[i]}}^{[i],O})] - \mu$ is the product cross-term of sensor i.

For comparison's sake, the following proposition gives the expanded expression of the product approximation:

Proposition 6.3. *The expanded expression of the product approximation is given by:*

$$\begin{aligned} D_{k+1|k+1}(x) \simeq & \left[\prod_{k=1}^{N}\left(1 - p_d^{[k]}(x)\right) + \sum_{1 \leqslant k^1 \leqslant N}\left(\sum_{j_1=1}^{m[k^1]} \frac{p_d^{[k^1]}(x) L_{z_{j_1}^{[k^1]}}^{[k^1]}(x). \prod_{k \neq k^1}\left(1 - p_d^{[k]}(x)\right)}{Den_P^{[k^1]}(z_{j_1}^{[k^1]})}\right)\right. \\ & \left. + \cdots + \sum_{1 \leqslant k^1 \leqslant \cdots \leqslant k^N \leqslant N}\left(\sum_{j_1=1}^{m[k^1]} \cdots \sum_{j_N=1}^{m[k^N]} \frac{p_d^{[k^1]}(x) L_{z_{j_1}^{[k^1]}}^{[k^1]}(x) \cdots p_d^{[k^N]}(x) L_{z_{j_N}^{[k^N]}}^{[k^N]}(x)}{Den_P^{[k^1]}(z_{j_1}^{[k^1]}) \cdots Den_P^{[k^1]}(z_{k_j^N}^{[k^N]})}\right)\right] D_{k+1|k}(x) \end{aligned} \tag{67}$$

where the product denominator of the i-th sensor $Den_P^{[i]}$ is given by:

$$Den_P^{[i]}(z) = \lambda^{[i]} c^{[i]}(z) + \mu s[p_d^{[i]} L_z^{[i]}] \tag{68}$$

The proof is easily obtained when expanding (65).

As it is shown on a simple example later in this document, the expanded expression of the product approximation (67) is similar to the multi-sensor data update expression (41). The numerators in (67) denote all the association combinations between an eventual target in x and the measurements by all the sensors (see example (81)). The exact same numerators are found in the expanded expression of the "exact" mutli-sensor data update (see example (80)). However, the cost of the independant processing of the sensors in the product approximation is clearly seen in the expression of the denominators (68), which fails at taking into account each association combination since no "multi-sensor normalization terms" such as $\mu s[p_d^{[1]} L_{z_1^{[1]}}^{[1]} p_d^{[2]} L_{z_1^{[2]}}^{[2]} (1 - p_d^{[3]}(x)) \cdots (1 - p_d^{[N]}(x))]$ appear.

6.3.2 Myopic sequential approximation

The second approximation is the more classic *myopic sequential approximation*, usually known as the iterated-corrector approximation [5], defined by:

Definition 6.4. *Assuming that the time updated density is Poisson with PHD $D_{k+1|k}(x) = \mu s(x)$, the sequence of intermediate myopic PHDs $\mu^{[i]} s^{[i]}(x)$ are given by:*

$$\forall i \in [1\ N], \mu^{[i]} s^{[i]}(x) = C_M^{[i]}(Z_{k+1}^{[i]}|x)\mu^{[i-1]} s^{[i-1]}(x) \tag{69}$$

where:

- $C_M^{[i]}(Z_{k+1}^{[i]}|x) = 1 - p_d^{[i]} + \sum_{z \in Z_{k+1}^{[i]}} \frac{p_d^{[i]}(x) L_z^{[i]}(x)}{\lambda^{[i]} c^{[i]}(z) + \mu^{[i-1]} s^{[i-1]}[p_d^{[i]} L_z^{[i]}]}$ *is the myopic corrector of the i-th sensor;*
- $\mu^{[0]} s^{[0]}(x) = \mu s(x) = D_{k+1|k}(x)$ *by convention.*

Then, the myopic sequential approximation is simply given by the last intermediate myopic PHD:

$$D_{k+1|k+1}(x) \simeq \mu^{[N]} s^{[N]}(x) \tag{70}$$

Again, the expression of the myopic corrector is similar to the single-sensor PHD data update equation (36) since $C_M^{[i]}$ is computed by ignoring all the measurements but those produced by sensor i. The major difference with the product approximation is that the myopic correctors $C_M^{[i]}$ can be contructed *recursively* and we have:

$$C_M^{[i]}(Z_{k+1}^{[i]}|x)\mu^{[i-1]} s^{[i-1]}(x) = \frac{\left[\frac{\delta}{\delta x}\left(\frac{\delta^{m[i]}}{\delta z_1^{[i]} \ldots \delta z_{m[i]}^{[i]}} e^{\beta_M^{[i]}[g^{[i]},h]}\right)\right]_{g^{[i]}=0,h=1}}{\left[\frac{\delta^{m[i]}}{\delta z_1^{[i]} \ldots \delta z_{m[i]}^{[i]}} e^{\beta_M^{[i]}[g^{[i]},h]}\right]_{g^{[i]}=0,h=1}} \tag{71}$$

where $\beta_M^{[i]}[g^{[i]}, h] = \lambda^{[i]} c^{[i]}[g^{[i]}] - \lambda^{[i]} + \mu^{[i-1]} s^{[i-1]}[h(1 - p_d^{[i]} + p_d^{[i]}.p_{g^{[i]}}^{[i],O})] - \mu^{[i-1]}$ is the myopic cross-term of sensor i. Unlike the product cross-terms (see (66)), the myopic cross-terms take into account the updated information resulting from the process of previous sensors. However, they ignore the fact that the current measurements from the remaining sensors are yet to be processed, hence their name.

For comparison's sake, the following proposition gives the expanded expression of the myopic sequential approximation:

Proposition 6.4. *The expanded expression of the myopic sequential approximation is given by:*

$$\begin{aligned} D_{k+1|k+1}(x) \simeq & \left[\prod_{k=1}^{N} \left(1 - p_d^{[k]}(x)\right) + \sum_{1 \leqslant k^1 \leqslant N} \left(\sum_{j_1=1}^{m[k^1]} \frac{p_d^{[k^1]}(x) L_{z_{j_1}^{[k^1]}}^{[k^1]}(x). \prod_{k \neq k^1} \left(1 - p_d^{[k]}(x)\right)}{Den_M^{[k^1]}(z_{j_1 s}^{[k^1]})} \right) \right. \\ & \left. + \cdots + \sum_{1 \leqslant k^1 \leqslant \cdots \leqslant k^N \leqslant N} \left(\sum_{j_1=1}^{m[k^1]} \cdots \sum_{j_N=1}^{m[k^N]} \frac{p_d^{[k^1]}(x) L_{z_{j_1}^{[k^1]}}^{[k^1]}(x) \cdots p_d^{[k^N]}(x) L_{z_{j_N}^{[k^N]}}^{[k^N]}(x)}{Den_M^{[k^1]}(z_{j_1}^{[k^1]}) \cdots Den_M^{[k^N]}(z_{j_N}^{[k^N]})} \right) \right] D_{k+1|k}(x) \end{aligned} \tag{72}$$

where the myopic denominator of the i-th sensor $Den_M^{[i]}$ is given by:

$$Den_M^{[i]}(z) = \lambda^{[i]} c^{[i]}(z) + \mu s \left[\prod_{k<i} \left(1 - p_d^{[k]}\right) .p_d^{[i]} L_z^{[i]} \right] + \sum_{1 \leqslant k^1 < i} \left(\sum_{j_1=1}^{m[k^1]} \frac{\mu s \left[\prod_{k<i, k \neq k^1} \left(1 - p_d^{[k]}\right) .p_d^{[k^1]} L_{z_{j_1}^{[k^1]}}^{[k^1]} p_d^{[i]} L_z^{[i]} \right]}{Den_M^{[k^1]}(z_{j_1}^{[k^1]})} \right)$$
$$+ \cdots + \sum_{1 \leqslant k^1 \leqslant \cdots \leqslant k^{i-1} < i} \left(\sum_{j_1=1}^{m[k^1]} \cdots \sum_{j_{i-1}=1}^{m[k^{i-1}]} \frac{\mu s \left[p_d^{[k^1]} L_{z_{j_1}^{[k^1]}}^{[k^1]} \cdots p_d^{[k^{i-1}]} L_{z_{j_{i-1}}^{[k^{i-1}]}}^{[k^{i-1}]} p_d^{[i]} L_z^{[i]} \right]}{Den_M^{[k^1]}(z_{j_1}^{[k^1]}) \cdots Den_M^{[k^{i-1}]}(z_{j_{i-1}}^{[k^{i-1}]})} \right) \quad (73)$$

The proof is given in section 8.20.

Once again, the expanded expression of the myopic approximation (72) is similar to the multi-sensor data update expression (41). The numerators in the expanded expression (see example (82)) are identical to those in the expanded expression of the "exact" multi-sensor data update (see example (80)) or those in the expanded expression of the product approximation (see example (81)). Unlike the product approximation, the myopic approximation does take into account the processing of the *previous* sensors; in some denominators "multi-sensor normalization terms" such as $\mu s \left[(1 - p_d^{[1]}) \cdots (1 - p_d^{[N-1]}) p_d^{[N]} L_z^{[N]}\right]$ appear. However, the *remaining* sensors (i.e. those whose measurements have not been processed yet) are ignored, hence the *myopic*. That is why some "multi-sensor normalization terms" such as $\mu s \left[(1 - p_d^{[1]}) \cdots (1 - p_d^{[i-1]}) p_d^{[i]} L_z^{[i]} (1 - p_d^{[i+1]}) \cdots (1 - p_d^{[N]})\right]$ will appear in the myopic approximation on a "truncated" form $\mu s \left[(1 - p_d^{[1]}) \cdots (1 - p_d^{[i-1]}) p_d^{[i]} L_z^{[i]}\right]$.

6.3.3 Nonmyopic sequential approximation

The third proposed approximation is the *non myopic sequential approximation*, defined by:

Definition 6.5. *Assuming that the time updated density is Poisson with PHD $D_{k+1|k}(x) = \mu s(x)$, the sequence of intermediate nonmyopic PHDs $\mu^{[i]} s^{[i]}(x)$ is given by:*

$$\forall i \in [1\ N], \mu^{[i]} s^{[i]}(x) = C_N^{[i]}(Z_{k+1}^{[i]}|x)\mu^{[i-1]} s^{[i-1]}(x) \tag{74}$$

where:

- $C_N^{[i]}(Z_{k+1}^{[i]}|x) = 1 - p_d^{[i]} + \sum_{z \in Z_{k+1}^{[i]}} \frac{p_d^{[i]}(x) L_z^{[i]}(x)}{\lambda^{[i]} c^{[i]}(z) + \mu^{[i-1]} s^{[i-1]} [p_d^{[i]} L_z^{[i]} (1 - p_d^{[i+1]}) \cdots (1 - p_d^{[N]})]}$ *is the non-myopic corrector of the i-th sensor;*
- $D_{k+1|k}(x) = \mu s(x) = \mu^{[0]} s^{[0]}(x)$ *by convention.*

Then, the nonmyopic sequential approximation is simply given by the last intermediate nonmyopic PHD:

$$D_{k+1|k+1}(x) \simeq \mu^{[N]} s^{[N]}(x) \tag{75}$$

Once more, the expression of the nonmyopic corrector is similar to the single-sensor PHD data update equation (36) since $C_N^{[i]}$ is computed by ignoring all the measurements but those produced by sensor i. The nonmyopic correctors $C_M^{[i]}$ can also be constructed *recursively* and we have:

$$C_N^{[i]}(Z_{k+1}^{[i]}|x)(1-p_d^{[i+1]}(x)) \cdots (1-p_d^{[N]}(x))\mu^{[i-1]} s^{[i-1]}(x) = \frac{\left[\frac{\delta}{\delta x}\left(\frac{\delta^{m[i]}}{\delta z_1^{[i]} \ldots \delta z_{m[i]}^{[i]}} e^{\beta_N^{[i]}[g^{[i]}, \cdots, g^{[N]}, h]}\right)\right]_{g^{[i]}=0, \cdots, g^{[N]}=0, h=1}}{\left[\frac{\delta^{m[i]}}{\delta z_1^{[i]} \ldots \delta z_{m[i]}^{[i]}} e^{\beta_N^{[i]}[g^{[i]}, \cdots, g^{[N]}, h]}\right]_{g^{[i]}=0, \cdots, g^{[N]}=0, h=1}} \tag{76}$$

where $\beta_N^{[i]}[g^{[i]}, \cdots, g^{[N]}, h] = \sum_{k=i}^{N} (\lambda^{[k]} c^{[k]}[g^{[k]}] - \lambda^{[k]}) + \mu^{[i-1]} s^{[i-1]} [h \prod_{k=i}^{N} (1 - p_d^{[k]} + p_d^{[k]} . p_{g^{[k]}}^{[k],O})] - \mu^{[i-1]}$ is the nonmyopic cross-term of sensor i. Unlike the product cross-terms (see (66)), the nonmyopic cross-terms take into account the updated information resulting from the process of previous sensors. Unlike the myopic cross-terms, they *also* take into the yet-to-be-processed sensors, hence their name.

For comparison's sake, the following proposition gives the expanded expression of the nonmyopic sequential approximation:

Proposition 6.5. *The expanded expression of the nonmyopic sequential approximation is given by:*

$$\begin{aligned} D_{k+1|k+1}(x) \simeq & \left[\prod_{k=1}^{N} \left(1 - p_d^{[k]}(x)\right) + \sum_{1 \leqslant k^1 \leqslant N} \left(\sum_{j_1=1}^{m[k^1]} \frac{\prod_{k \neq k^1} \left(1 - p_d^{[k]}(x)\right) . p_d^{[k^1]}(x) L_{z_{j_1}^{[k^1]}}^{[k^1]}(x)}{Den_N^{[k^1]}(z_{j_1}^{[k^1]})} \right) \right. \\ & \left. + \cdots + \sum_{1 \leqslant k^1 \leqslant \cdots \leqslant k^N \leqslant N} \left(\sum_{j_1=1}^{m[k^1]} \cdots \sum_{j_N=1}^{m[k^N]} \frac{p_d^{[k^1]}(x) L_{z_{j_1}^{[k^1]}}^{[k^1]}(x) \cdots p_d^{[k^N]}(x) L_{z_{j_N}^{[k^N]}}^{[k^N]}(x)}{Den_N^{[k^1]}(z_{j_1}^{[k^1]}) \cdots Den_N^{[k^N]}(z_{j_N}^{[k^N]})} \right) \right] D_{k+1|k}(x) \end{aligned} \tag{77}$$

where the nonmyopic denominator of the i-th sensor $Den_N^{[i]}$ is given by:

$$
\begin{aligned}
Den_N^{[i]}(z) =& \lambda^{[i]} c^{[i]}(z) + \mu s \left[\prod_{k \neq i} \left(1 - p_d^{[k]}\right) . p_d^{[i]} L_z^{[i]} \right] + \sum_{1 \leqslant k^1 < i} \left(\sum_{j_1=1}^{m[k^1]} \frac{\mu s \left[\prod_{k \neq i, k \neq k^1} \left(1 - p_d^{[k]}\right) . p_d^{[k^1]} L^{[k^1]}_{z^{[k^1]}_{j_1}} p_d^{[i]} L_z^{[i]} \right]}{Den_N^{[k^1]}(z^{[k^1]}_{j_1})} \right) \\
& + \cdots + \sum_{1 \leqslant k^1 \leqslant \cdots \leqslant k^{i-1} < i} \left(\sum_{j_1=1}^{m[k^1]} \cdots \sum_{j_{i-1}=1}^{m[k^{i-1}]} \frac{\mu s \left[\prod_{k > i} \left(1 - p_d^{[k]}\right) . p_d^{[k^1]} L^{[k^1]}_{z^{[k^1]}_{j_1}} \cdots p_d^{[k^{i-1}]} L^{[k^{i-1}]}_{z^{[k^{i-1}]}_{j_{i-1}}} p_d^{[i]} L_z^{[i]} \right]}{Den_N^{[k^1]}(z^{[k^1]}_{j_1}) \cdots Den_N^{[k^{i-1}]}(z^{[k^{i-1}]}_{j_{i-1}})} \right)
\end{aligned}
\tag{78}
$$

The proof is similar to the myopic case (see section 8.20).

Again, the expanded expression of the nonmyopic approximation (77) is similar to the multi-sensor data update expression (41). The numerators in the expanded expression (see example (83)) are identical to those in the expanded expression of the "exact" multi-sensor data update (see example (80)), those in the expanded expression of the product (see example (81)) or the myopic approximation (see example (82)). Unlike the myopic case, no "multi-sensor normalization terms" such as $\mu s \left[(1 - p_d^{[1]}) \cdots (1 - p_d^{[i-1]}) p_d^{[i]} L_z^{[i]} (1 - p_d^{[i+1]}) \cdots (1 - p_d^{[N]})\right]$ appear in the nonmyopic approximation on a "truncated" form $\mu s \left[(1 - p_d^{[1]}) \cdots (1 - p_d^{[i-1]}) p_d^{[i]} L_z^{[i]}\right]$. In that sense, the nonmyopic approximation seems closer to the "exact" multi-sensor data update than the myopic approximation.

6.3.4 Elements of comparison between the three approximation methods

Although these approximations are currently implemented with particle filtering methods, no Monte Carlo runs have been implemented yet; thus it is fairly difficult to compare the three approximations with regard to the "exact" multi-sensor data update, concerning their complexity as well as their accuracy. The current version of this document aims at describing these approximations rather than testing and comparing them, this next step will nevertheless be necessary in further studies.

The easier aspect on which these approximations can be compared is perhaps the complexity. Assume that, at current step, each sensor i produces an average of M measurements. Assume also that the PHD is implemented with a particle filter, i.e. the time updated density $D_{k+1|k}$ is approximated by a collection of weighted particles $\{w_j, x_j\}$, the current number of particles being K. Recall that in the single-sensor data update (see (36)), the generic cross-term has one obervation functional g and one state fonctional h. Therefore, the generic cross-term can be set in any one of the M measurements of the current sensor and/or any one of the K particles; the number of cross-term to compute is roughly $K.M$. By construction, the three approximations are merely sequences of N single-sensor data updates; indeed, the construction of each corrector (see definitions 6.3, 6.4 and 6.5) is similar to a single-sensor data update (see theorem 4.2). The number of cross-terms N_β^{approx} for each approximation roughly $N.K.M$ and we have:

$$N_\beta^{approx} \sim \mathcal{O}(N.K.M) \tag{79}$$

Even though the approximation methods spare the computation of a large number of cross-terms (compare the result above with result (62)), the most significant computational gain with the approximations is likely to stem from the fact that the correctors have nice tractable expressions, whereas the multi-sensor Bayes equation do not. That is, the computional burden of the generation of combinational terms in the general case (see example (43)) is unnecessary in the computation of the correctors whether product, myopic or nonmyopic. However, the number of computed cross-terms is a limited criteria since it does not account for the fact the cross-terms are *different* in each approximation methods, which may have an effect on the complexity as well.

Comparing the accuracy of the three approximation methods through their closed-form expressions (67), (72), (77) seems quite challenging. Should a closed-form expression of the "exact" multi-sensor data updated density (41) were available, one could use this density as a reference density to compare the three approximated PHDs with the following criteria:

1. The expected number of targets according to each PHD;
2. The Kullback-Leibler distance between the normalized reference density and each normalized PHD.

Even if the closed-form expression of the reference density is not available, the same criteria should be computed on Monte Carlo runs in further studies in order to be able to quantify the accuracy of each one of the approximations. The following paragraph is by no means a rigorous and quantitative comparison, it merely shows the differences between the four PHDs (the "exact" one and the three approximations) on a very simple example.

Suppose that there are two sensors, each one produced a single measurement in the current time (i.e. $Z^{[1]}_{k+1} = \{z^{[1]}_1\}$ and $Z^{[2]}_{k+1} = \{z^{[2]}_1\}$). For clarity'sake, $1 - p^{[1]}_d(x)$ and $1 - p^{[2]}_d(x)$ will be respectively denoted by $q^{[1]}_d(x)$ and $q^{[2]}_d(x)$. Furthermore, since there is only one measurement per sensor, there is no association ambiguity and therefore $c^{[1]}(z^{[1]}_1)$, $L^{[1]}_{z^{[1]}_1}(x)$, $c^{[2]}(z^{[2]}_1)$, $L^{[2]}_{z^{[2]}_1}(x)$ will be respectively denoted by $c^{[1]}$, $L^{[1]}$, $c^{[2]}$ and $L^{[2]}$. The expression of the data update correctors in a given $x \in \mathcal{X}$ (i.e. the ratio of the data updated PHD $D_{k+1|k+1}(x)$ over the time updated corrector $D_{k+1|k}(x)$) for each method are:

First, the "reference corrector" is provided by the multi-sensor data update:

$$
\begin{aligned}
\frac{D_{k+1|k+1}(x)}{D_{k+1|k}(x)} &= q^{[1]}_d(x) q^{[2]}_d(x) \\
&+ \frac{p^{[1]}_d(x) L^{[1]}(x) q^{[2]}_d(x)}{\lambda^{[1]} c^{[1]} + \mu s[p^{[1]}_d L^{[1]} q^{[2]}_d] + \frac{\mu s[p^{[1]}_d L^{[1]} p^{[2]}_d L^{[2]}]}{\lambda^{[2]} c^{[2]} + \mu s[q^{[1]}_d p^{[2]}_d L^{[2]}]}} \\
&+ \frac{q^{[1]}_d(x) p^{[2]}_d(x) L^{[2]}(x)}{\lambda^{[2]} c^{[2]} + \mu s[q^{[1]}_d p^{[2]}_d L^{[2]}] + \frac{\mu s[p^{[1]}_d L^{[1]} p^{[2]}_d L^{[2]}]}{\lambda^{[1]} c^{[1]} + \mu s[p^{[1]}_d L^{[1]} q^{[2]}_d]}} \\
&+ \frac{p^{[1]}_d(x) L^{[1]}(x) p^{[2]}_d(x) L^{[2]}(x)}{(\lambda^{[1]} c^{[1]} + \mu s[p^{[1]}_d L^{[1]} q^{[2]}_d])(\lambda^{[2]} c^{[2]} + \mu s[q^{[1]}_d p^{[2]}_d L^{[2]}]) + \mu s[p^{[1]}_d L^{[1]} p^{[2]}_d L^{[2]}]}
\end{aligned}
\tag{80}
$$

Then, the product approximation gives:

$$\begin{aligned}\frac{D_{k+1|k+1}(x)}{D_{k+1|k}(x)} \simeq\ & q_d^{[1]}(x)q_d^{[2]}(x)\\ &+ \frac{p_d^{[1]}(x)L^{[1]}(x)q_d^{[2]}(x)}{\lambda^{[1]}c^{[1]} + \mu s[p_d^{[1]}L^{[1]}]}\\ &+ \frac{q_d^{[1]}(x)p_d^{[2]}(x)L^{[2]}(x)}{\lambda^{[2]}c^{[2]} + \mu s[p_d^{[2]}L^{[2]}]}\\ &+ \frac{p_d^{[1]}(x)L^{[1]}(x)p_d^{[2]}(x)L^{[2]}(x)}{(\lambda^{[1]}c^{[1]} + \mu s[p_d^{[1]}L^{[1]}])(\lambda^{[2]}c^{[2]} + \mu s[p_d^{[2]}L^{[2]}])}\end{aligned} \tag{81}$$

Then, the myopic sequential approximation gives:

$$\begin{aligned}\frac{D_{k+1|k+1}(x)}{D_{k+1|k}(x)} \simeq\ & q_d^{[1]}(x)q_d^{[2]}(x)\\ &+ \frac{p_d^{[1]}(x)L^{[1]}(x)q_d^{[2]}(x)}{\lambda^{[1]}c^{[1]} + \mu s[p_d^{[1]}L^{[1]}]}\\ &+ \frac{q_d^{[1]}(x)p_d^{[2]}(x)L^{[2]}(x)}{\lambda^{[2]}c^{[2]} + \mu s[q_d^{[1]}p_d^{[2]}L^{[2]}] + \frac{\mu s[p_d^{[1]}L^{[1]}p_d^{[2]}L^{[2]}]}{\lambda^{[1]}c^{[1]}+\mu s[p_d^{[1]}L^{[1]}]}}\\ &+ \frac{p_d^{[1]}(x)L^{[1]}(x)p_d^{[2]}(x)L^{[2]}(x)}{(\lambda^{[1]}c^{[1]} + \mu s[p_d^{[1]}L^{[1]}])(\lambda^{[2]}c^{[2]} + \mu s[q_d^{[1]}p_d^{[2]}L^{[2]}]) + \mu s[p_d^{[1]}L^{[1]}p_d^{[2]}L^{[2]}]}\end{aligned} \tag{82}$$

Finally, the nonmyopic sequential approximation gives:

$$\begin{aligned}\frac{D_{k+1|k+1}(x)}{D_{k+1|k}(x)} \simeq\ & q_d^{[1]}(x)q_d^{[2]}(x)\\ &+ \frac{p_d^{[1]}(x)L^{[1]}(x)q_d^{[2]}(x)}{\lambda^{[1]}c^{[1]} + \mu s[p_d^{[1]}L^{[1]}q_d^{[2]}]}\\ &+ \frac{q_d^{[1]}(x)p_d^{[2]}(x)L^{[2]}(x)}{\lambda^{[2]}c^{[2]} + \mu s[q_d^{[1]}p_d^{[2]}L^{[2]}] + \frac{\mu s[p_d^{[1]}L^{[1]}p_d^{[2]}L^{[2]}]}{\lambda^{[1]}c^{[1]}+\mu s[p_d^{[1]}L^{[1]}q_d^{[2]}]}}\\ &+ \frac{p_d^{[1]}(x)L^{[1]}(x)p_d^{[2]}(x)L^{[2]}(x)}{(\lambda^{[1]}c^{[1]} + \mu s[p_d^{[1]}L^{[1]}q_d^{[2]}])(\lambda^{[2]}c^{[2]} + \mu s[q_d^{[1]}p_d^{[2]}L^{[2]}]) + \mu s[p_d^{[1]}L^{[1]}p_d^{[2]}L^{[2]}]}\end{aligned} \tag{83}$$

The first expression (80) is simply the multi-sensor data update equation (41) applied to this example; it is the reference corrector in the sense that one would like the approximation to be as close as possible to the multi-sensor data update equation. The last denominator is a normalization over all the state space of the association likelihoods, it denote the likelihood that the two current measurements are produced, either by two different sources $((\lambda^{[1]}c^{[1]} + \mu s[p_d^{[1]}L^{[1]}q_d^{[2]}])(\lambda^{[2]}c^{[2]} + \mu s[q_d^{[1]}p_d^{[2]}L^{[2]}]))$ or by a single source $(\mu s[p_d^{[1]}L^{[1]}p_d^{[2]}L^{[2]}])$. Modifying each fraction such they have this same denominator provides an intuitive interpretation of the corrector. For example, the second term becomes:

$$\frac{p_d^{[1]}(x)L^{[1]}(x)q_d^{[2]}(x).(\lambda^{[2]}c^{[2]} + \mu s[q_d^{[1]}p_d^{[2]}L^{[2]}])}{(\lambda^{[1]}c^{[1]} + \mu s[p_d^{[1]}L^{[1]}q_d^{[2]}]).(\lambda^{[2]}c^{[2]} + \mu s[q_d^{[1]}p_d^{[2]}L^{[2]}]) + \mu s[p_d^{[1]}L^{[1]}p_d^{[2]}L^{[2]}]}$$

which denote the likelihood that, given the two current measurements, the (eventual) target in x produced the measurement $z_1^{[1]}$ *but* was undetected by the second sensor ($p_d^{[1]}(x)L^{[1]}(x)q_d^{[2]}(x)$), whereas the second measurement $z_1^{[2]}$ was produced by *another* source ($\lambda^{[2]}c^{[2]} + \mu s[q_d^{[1]}p_d^{[2]}L^{[2]}]$).

A quick glance to the approximated correctors shows that the four numerators are identical to those in the reference (80), that is, the four possible target in x-to-measurement associations are covered. However, because the measurements are processed *sequentially* (per sensor) rather than *in a whole*, each approximation fails at enumerating all the measurement-to-target associations in the normalization terms, hence the differences between the denominators in the approximated correctors and in the reference corrector.

In the product corrector (81), the batch of measurements of each sensor are processed independently; therefore, the associations of a single source to measurements from different sensors are never covered. In the example above, it is clearly shown since the multi-sensor cross-term $\mu s[p_d^{[1]}L^{[1]}p_d^{[2]}L^{[2]}]$ does not appear. Furthermore, cross-terms such as $\lambda^{[2]}c^{[2]} + \mu s[q_d^{[1]}p_d^{[2]}L^{[2]}]$ are approximated by $\lambda^{[2]}c^{[2]} + \mu s[p_d^{[2]}L^{[2]}]$ which fails at considering multi-sensor interaction. Note the expression of the product corrector is symmetrical with respect to both sensors. The strong advantage of the product corrector lies indeed in the fact that the resulting density does not depend on the order in which the sensors are processed.

In the myopic corrector case (82), the sensors are still processed sequentially but at each step, the input is the density resulting from the processing of the previous sensors rather then the time updated density. In the example above, it means that the processing of the first sensor is equivalent in both product and myopic correctors (the ratio $\frac{p_d^{[1]}(x)L^{[1]}(x)q_d^{[2]}(x)}{\lambda^{[1]}c^{[1]}+\mu s[p_d^{[1]}L^{[1]}]}$ appears in both (81) and (82)), but the processing of the second sensor in the myopic case does include multi-sensor interaction with the first sensor, whereas the product approximation does not. Note that the approximated normalization $\lambda^{[1]}c^{[1]} + \mu s[p_d^{[1]}L^{[1]}]$ (rather than $\lambda^{[1]}c^{[1]} + \mu s[p_d^{[1]}L^{[1]}q_d^{[2]}]$) does appear in subsequent denominators in the myopic case. Indeed, the denominators are constructed recursively (see equation (73)) and initial approximations *propagate* through the sequence of denominators. On this example, the myopic corrector seems "closer" to the reference than the product one is, but this proves by no means the superior accuracy of the myopic method. Note that the myopic corrector is inherently *asymmetrical* with respect to the sensors, the order in which the sensors are processed could be an important factor contributing to its relative accuracy.

The nonmyopic method (83) aims at smoothing the successive approximations propagated through the denominator of the myopic corrector. In the example above, the cross-term $\lambda^{[1]}c^{[1]} + \mu s[p_d^{[1]}L^{[1]}q_d^{[2]}]$ which is computed during the processing of the first sensor *does* take into account the second sensor as well. Since this "exact" cross-term is propagated in subsequent denominators, the nonmyopic corrector seems event "closer" to the reference that the myopic corrector is. Note that the nonmyopic case shares the assymmetry issue with the myopic one.

7 Conclusion

This section will be filled when further work will allow the comparison of the several proposed extensions through proper simulations.

8 Mathematical proofs

8.1 Property (13)

Proof.

$$\begin{aligned}
\frac{\delta G}{\delta x}[h] &= \frac{\partial G}{\partial \delta_x}[h] && (using\ (11))\\
&= \lim_{\epsilon \to 0} \frac{G[h+\epsilon.\delta_x]-G[h]}{\epsilon} && (using\ (8))\\
&= \lim_{\epsilon \to 0} \frac{h(x_1)+\epsilon.\delta_x(x_1)-h(x_1)}{\epsilon}\\
&= \delta_x(x_1)
\end{aligned}$$

□

8.2 Property (14)

Proof.

$$\begin{aligned}
\frac{\delta G}{\delta x}[h] &= \frac{\partial G}{\partial \delta_x}[h] && (using\ (11))\\
&= \lim_{\epsilon \to 0} \frac{G[h+\epsilon.\delta_x]-G[h]}{\epsilon} && (using\ (8))\\
&= \lim_{\epsilon \to 0} \frac{\int (h(x)+\epsilon.\delta_x(x_1))f(x)dx - \int h(x)f(x)dx}{\epsilon}\\
&= f(x_1)
\end{aligned}$$

□

8.3 Proposition 3.1

Proof.

$$\begin{aligned}
\frac{\delta^n G_\Xi}{\delta x_1 ... \delta x_n}[h] &= \int \frac{\delta^n h^Y}{\delta x_1 ... \delta x_n} f_\Xi(Y)\delta Y && (using\ (5))\\
&= \sum_{p=n}^{\infty} \frac{1}{p!}\int_{\mathcal{X}^p} \sum_{1 \leqslant i_1 \neq ... \neq i_n \leqslant p} \left[h(y_1)...\frac{\delta h}{\delta x_1}(y_{i_1})...\frac{\delta h}{\delta x_n}(y_{i_n})...h(y_p)\right] j_{\Xi,p}(y_1,...y_p)dy_1...dy_p\\
&= \sum_{p=n}^{\infty} \frac{1}{p!}\int_{\mathcal{X}^p} \sum_{1 \leqslant i_1 \neq ... \neq i_n \leqslant p} [h(y_1)...\delta_{x_1}(y_{i_1})...\delta_{x_n}(y_{i_n})...h(y_p)]\, j_{\Xi,p}(y_1,...y_p)dy_1...dy_p && (using\ (13))
\end{aligned}$$

Setting h to 0 gives:

$$\begin{aligned}
\frac{\delta^n G_\Xi}{\delta x_1 ... \delta x_n}[0] &= \frac{1}{n!}\int_{\mathcal{X}^n} \sum_{1 \leqslant i_1 \neq ... \neq i_n \leqslant n} [\delta_{x_1}(y_{i_1})...\delta_{x_n}(y_{i_n})]\, j_{\Xi,n}(y_1,...y_n)dy_1...dy_n\\
&= \int_{\mathcal{X}^n} [\delta_{x_1}(y_1)...\delta_{x_n}(y_n)]\, j_{\Xi,n}(y_1,...y_n)dy_1...dy_n\\
&= j_{\Xi,n}(x_1,...x_n)\\
&= f_\Xi(X)
\end{aligned}$$

□

8.4 Theorem 3.1

Proof.

$$\frac{\delta^n G_\Xi}{\delta x_1 ... \delta x_n}[h] = \int \frac{\delta^n h^Y}{\delta x_1 ... \delta x_n} f_\Xi(Y)\delta Y \qquad (using\ (5))$$

$$= \sum_{p=n}^{\infty} \frac{1}{p!} \int_{\mathcal{X}^p} \sum_{1 \leqslant i_1 \neq ... \neq i_n \leqslant p} \left[h(y_1) ... \frac{\delta h}{\delta x_1}(y_{i_1}) ... \frac{\delta h}{\delta x_n}(y_{i_n}) ... h(y_p) \right] j_{\Xi,p}(y_1, ... y_p) dy_1 ... dy_p$$

$$= \sum_{p=n}^{\infty} \frac{1}{p!} \int_{\mathcal{X}^p} \sum_{1 \leqslant i_1 \neq ... \neq i_n \leqslant p} \left[h(y_1) ... \delta_{x_1}(y_{i_1}) ... \delta_{x_n}(y_{i_n}) ... h(y_p) \right] j_{\Xi,p}(y_1, ... y_p) dy_1 ... dy_p \qquad (using\ (13))$$

Setting h to 1 gives:

$$\frac{\delta^n G_\Xi}{\delta x_1 ... \delta x_n}[1] = \sum_{p=n}^{\infty} \frac{1}{p!} \int_{\mathcal{X}^p} \sum_{1 \leqslant i_1 \neq ... \neq i_n \leqslant p} \left[\delta_{x_1}(y_{i_1}) ... \delta_{x_n}(y_{i_n}) \right] j_{\Xi,p}(y_1, ... y_p) dy_1 ... dy_p$$

$$= \sum_{p=n}^{\infty} \frac{A_p^n}{p!} \int_{\mathcal{X}^p} \left[\delta_{x_1}(y_1) ... \delta_{x_n}(y_n) \right] j_{\Xi,p}(y_1, ... y_p) dy_1 ... dy_p$$

$$= \sum_{p=n}^{\infty} \frac{1}{(p-n)!} \int_{\mathcal{X}^{p-n}} j_{\Xi,p}(x_1, ..., x_n, w_1, ..., w_{p-n}) dw_1 ... dw_{p-n}$$

$$= \sum_{p=0}^{\infty} \frac{1}{p!} \int_{\mathcal{X}^p} j_{\Xi,p+n}(x_1, ..., x_n, w_1, ..., w_p) dw_1 ... dw_p$$

$$= \int f_\Xi(\{x_1, ..., x_n\} \cup W) \delta W$$

$$= D_\Xi(X)$$

□

8.5 Proposition 3.2

Proof.

$$\int h_Y f_\Xi(Y) \delta Y = \sum_{n=0}^{\infty} \frac{1}{n!} \int_{\mathcal{X}^n} \left(\sum_{i=1}^{n} h(y_i) \right) j_{\Xi,n}(y_1, ..., y_n) dy_1 ... dy_n$$

$$= \sum_{n=0}^{\infty} \frac{A_n^1}{n!} \int_{\mathcal{X}^n} h(y_1) j_{\Xi,n}(y_1, ..., y_n) dy_1 ... dy_n$$

$$= \sum_{n=1}^{\infty} \frac{1}{(n-1)!} \int_{\mathcal{X}} h(y) \left(\int_{\mathcal{X}^{n-1}} j_{\Xi,n}(y, w_1, ..., w_{n-1}) dw_1 ... dw_{n-1} \right) dy$$

$$= \int_{\mathcal{X}} h(y) \left(\sum_{n=0}^{\infty} \frac{1}{n!} \int_{\mathcal{X}^n} j_{\Xi,n+1}(y, w_1, ..., w_n) dw_1 ... dw_n \right) dy$$

$$= \int_{\mathcal{X}} h(y) \left(\int f_\Xi(\{y\} \cup W) \delta W \right) dy$$

$$= \int_{\mathcal{X}} h(y) D_\Xi(\{y\}) dy$$

□

8.6 Theorem 3.3

Proof.

$$\begin{aligned}
D(\{x\}) &= \frac{\delta G}{\delta x}[1] && (using\ (17))\\
&= \int \left[\frac{\delta}{\delta x}\Phi[h]^W\right]_{h=1} f_\Xi(W)\delta W && (using\ (5))\\
&= \int \left[\sum_{w\in W}\left(\frac{\delta}{\delta x}\Phi[h](w).\Phi[h]^{W\setminus\{w\}}\right)\right]_{h=1} f_\Xi(W)\delta W\\
&= \int \left[\sum_{w\in W}\left(\frac{\delta \Phi_w}{\delta x}[h].\Phi[h]^{W\setminus\{w\}}\right)\right]_{h=1} f_\Xi(W)\delta W\\
&= \int \left(\sum_{w\in W}\frac{\delta \Phi_w}{\delta x}[1]\right) f_\Xi(W)\delta W\\
&= \int \sum_{w\in W} D_w(\{x\}) f_\Xi(W)\delta W && (using\ (17))
\end{aligned}$$

Setting $h_x(w) = D_w(\{x\})$ and using (18) gives:

$$\begin{aligned}
D(\{x\}) &= \int h_x(w) D_\Xi(\{w\})dw\\
&= \int D_w(\{x\}) D_\Xi(\{w\})dw
\end{aligned}$$

□

8.7 Theorem 3.2

Proof.

$$\begin{aligned}
\int_S D_{k|k}(\{x\}|Z^{(k)})dx &= \int 1_S(x) D_{k|k}(\{x\}|Z^{(k)})dx\\
&= \int 1_S(x)\left(\int \delta_Y(x) f_{k|k}(Y|Z^{(k)})\delta Y\right)dx && (using\ (19))\\
&= \int \left(\int 1_S(x)\delta_Y(x)dx\right) f_{k|k}(Y|Z^{(k)})\delta Y\\
&= \int |Y\cap S| f_{k|k}(Y|Z^{(k)})\delta Y\\
&= \mathbb{E}(|\Xi_{k|k}\cap S|)
\end{aligned}$$

But, according to the definition 2.1:

$$\int_S D_{k|k}(x|Z^{(k)})dx = \mathbb{E}(|\Xi_{k|k}\cap S|)$$

Thus, $\int_S D_{k|k}(x|Z^{(k)})dx = \int_S D_{k|k}(\{x\}|Z^{(k)})dx$ and therefore$D_{k|k}(x|Z^{(k)}) = D_{k|k}(\{x\}|Z^{(k)})$ almost everywhere. □

8.8 Proposition 4.1

Proof. Conditionnaly on the multi-target set $X = \{x_k^1, ..., x_k^n\}$ the RFS $\Xi_{k+1|k}$ equals to:

$$\Xi_{k+1|k} = \Xi^E_{k+1|k}(x_k^1) \cup ... \cup \Xi^E_{k+1|k}(x_k^n) \cup \Xi^S_{k+1|k}(x_k^1) \cup ... \cup \Xi^S_{k+1|k}(x_k^n) \cup \Xi^B_{k+1|k}$$

where:

- $\Xi^E_{k+1|k}(x_k^i)$ is the RFS describing the evolution of the i-th existing target;
- $\Xi^S_{k+1|k}(x_k^i)$ is the RFS describing the creation of spawning targets originating from the i-th existing target;
- $\Xi^B_{k+1|k}$ is the RFS describing the creation of spontaneous birth targets.

Since $\Xi^E_{k+1|k}(x_k^i)$ equals to:

- $\emptyset$ with probability $1 - p_s(x_k^i)$ (target i dies);
- $\{y\}, y \in \mathcal{X}$ with probability $f^E_{k+1|k}(y|x_k^i)p_s(x_k^i)$ (target i survives and evolves to new state y);
- $Y \in \mathcal{X}^n, n \geqslant 2$ with zero probability. [12]

its PGFl $G_{\Xi^E_{k+1|k}(x_k^i)}$ equals to:

$$\begin{aligned}
G_{\Xi^E_{k+1|k}(x_k^i)}[h] &= \int h^Y f^E_{k+1|k}(Y|x_k^i)\delta Y \\
&= \underbrace{j_{\Xi^E_{k+1|k}(x_k^i),0}(\emptyset)}_{=1-p_s(x_k^i)} + \int_{\mathcal{X}^1} h(y_1) \underbrace{j_{\Xi^E_{k+1|k}(x_k^i),1}(y_1)}_{=f^E_{k+1|k}(y_1|x_k^i)p_s(x_k^i)} dy_1 \\
&\quad + \sum_{p=2}^{\infty} \frac{1}{p!} \int_{\mathcal{X}^p} h(y_1)...h(y_p) \underbrace{j_{\Xi^E_{k+1|k}(x_k^i),p}(y_1,...,y_p)}_{=0} dy_1...dy_p \\
&= 1 - p_s(x_k^i) + \int h(y) f^E_{k+1|k}(y|x_k^i)p_s(x_k^i)dy \\
&= 1 - p_s(x_k^i) + p_s(x_k^i)p_h^E(x_k^i)
\end{aligned}$$

Denoting by $p_h^S(x_k^i) = \int h^Y f^S_{k+1|k}(Y|x_k^i)\delta Y$ the PGFl of $\Xi^S_{k+1|k}(x_k^i)$ and by $p_h^B = \int h^Y f^B_{k+1|k}(Y)\delta Y$ the PGFl of $\Xi^B_{k+1|k}$ and using the product rule of independent PGFls (7), we have:

$$\begin{aligned}
G_{k+1|k}[h|X] &= \prod_{x \in X} \left(G_{\Xi^E_{k+1|k}(x)}[h] \right) . \prod_{x \in X} \left(G_{\Xi^S_{k+1|k}(x)}[h] \right) .G_{\Xi^B_{k+1|k}}[h] \\
&= (1 - p_s + p_s.p_h^E)^X .(p_h^S)^X .p_h^B
\end{aligned}$$

$\square$

[12] Remember that "splitting" targets are *not* covered by the transition process but by the spawning process.

8.9 Theorem 4.1

Proof. Using the definition of the PGFl (5), we can write:

$$
\begin{aligned}
G_{k+1|k}[h] &= \int h^Y f_{k+1|k}(Y|Z^{(k)})\delta Y \\
&= \int h^Y \left(\int f_{k+1|k}(Y|X) f_{k|k}(X|Z^{(k)})\delta X \right) \delta Y && (\mathit{using}\ (1)) \\
&= \int \left(\int h^Y f_{k+1|k}(Y|X)\delta Y \right) f_{k|k}(X|Z^{(k)})\delta X \\
&= \int G_{k+1|k}[h|X] f_{k|k}(X|Z^{(k)})\delta X && (\mathit{using}\ (5)) \\
&= \int (1 - p_s + p_s.p_h^E)^X.(p_h^S)^X.p_h^B f_{k|k}(X|Z^{(k)})\delta X && (\mathit{using}\ (24)) \\
&= p_h^B. \int \left((1 - p_s + p_s.p_h^E).(p_h^S)\right)^X f_{k|k}(X|Z^{(k)})\delta X
\end{aligned}
$$

Let Φ be the functional $\Phi[h](x) = (1 - p_s(x) + p_s(x).p_h^E(x)).(p_h^S(x))$, then:

$$
\begin{aligned}
G_{k+1|k}[h] &= p_h^B. \int (\Phi[h])^X f_{k|k}(X|Z^{(k)})\delta X \\
&= p_h^B.G_{k|k}[\Phi[h]] && (\mathit{using}\ (5))
\end{aligned}
$$

Then, using the caracterization of the PHD as a set derivative of the PGFl (23) gives:

$$
\begin{aligned}
D_{k+1|k}(x) &= \left[\frac{\delta G_{k+1|k}}{\delta x}[h] \right]_{h=1} \\
&= \underbrace{\left[\frac{\delta p_h^B}{\delta x} \right]_{h=1}}_{A} . \underbrace{G_{k|k}[\Phi[1]]}_{B} + \underbrace{p_1^B}_{C} . \underbrace{\left[\frac{\delta G_{k|k}}{\delta x}[\Phi[h]] \right]_{h=1}}_{D}
\end{aligned}
$$

Since $b_{k+1|k}^B(x)$ is the PHD of the spontaneous birth multitarget RFS $\Xi_{k+1|k}^B$ with PGFl p_h^B, using (23) again gives:

$$
A = \left[\frac{\delta p_h^B}{\delta x} \right]_{h=1} = b_{k+1|k}^B(x)
$$

Then, one can note that, for all $x \in \mathcal{X}$:

$$
\begin{aligned}
\Phi[1](x) &= (1 - p_s(x) + p_s(x).p_1^E(x)).(p_1^S(x)) \\
&= \left(1 - p_s(x) + p_s(x). \int 1(y) f_{k+1|k}^E(y|x)dy \right) . \left(\int 1^Y f_{k+1|k}^S(Y|x_k^i)\delta Y \right) \\
&= \left(1 - p_s(x) + p_s(x). \underbrace{\int f_{k+1|k}^E(y|x)dy}_{=1} \right) . \left(\underbrace{\int f_{k+1|k}^S(Y|x_k^i)\delta Y}_{=1} \right) \\
&= 1 - p_s(x) + p_s(x) \\
&= 1
\end{aligned}
$$

Then, since $\Phi[1] = 1$:

$$\begin{aligned} B &= G_{k|k}[\Phi[1]] \\ &= G_{k|k}[1] \\ &= \int 1^Y f_{k|k}(Y|Z^{(k)})\delta Y && (\textit{using } (5)) \\ &= 1 \end{aligned}$$

Using once more the definition of a PGFl (5) gives:

$$C = p_1^B = \int 1^Y f^B_{k+1|k}(Y)\delta Y = 1$$

Finally, we have:

$$D = \left[\frac{\delta G_{k|k}}{\delta x}[\Phi[h]]\right]_{h=1} = \hat{D}_{k|k}(\{x\}) \qquad (\textit{using } (17))$$

where $\hat{D}_{k|k}$ denotes the moment density corresponding to the PGFl $G_{k|k}[\Phi[h]]$. Denote by Φ_w the PGFl $\Phi_w[h] = \Phi[h](w)$ and $\hat{D}_w$ the corresponding moment density. Then, since $\Phi[1] = 1$, proposition 3.3 applies and we have:

$$\begin{aligned} D &= \int \hat{D}_w(\{x\}) D_{k|k}(\{w\})dw \\ &= \int \left[\frac{\delta \Phi_w}{\delta x}[h]\right]_{h=1} D_{k|k}(\{w\})dw && (\textit{using } (17)) \\ &= \int \left[\frac{\delta \Phi_w}{\delta x}[h]\right]_{h=1} D_{k|k}(w)dw && (\textit{using } (22)) \end{aligned}$$

Then:

$$\begin{aligned} \frac{\delta \Phi_w}{\delta x}[h] &= \frac{\delta}{\delta x}\left[((1 - p_s(w) + p_s(w).p_h^E(w)).(p_h^S(w))\right] \\ &= p_s(w)\left(\frac{\delta}{\delta x}p_h^E(w)\right)p_h^S(w) + (1 - p_s(w) + p_s(w).p_h^E(w))\left(\frac{\delta}{\delta x}p_h^S(w)\right) \end{aligned}$$

Setting $G[h] = p_h^E(w) = \int h(y) f^E_{k+1|k}(y|w)dy$ and using property (14) gives:

$$\frac{\delta \Phi_w}{\delta x}[h] = p_s(w) f^E_{k+1|k}(x|w) p_h^S(w) + (1 - p_s(w) + p_s(w).p_h^E(w))\left(\frac{\delta}{\delta x}p_h^S(w)\right)$$

Therefore, by setting $h = 1$, one have:

$$\left[\frac{\delta \Phi_w}{\delta x}[h]\right]_{h=1} = p_s(w) f^E_{k+1|k}(x|w) p_1^S(w) + (1 - p_s(w) + p_s(w).p_1^E(w))\left[\frac{\delta}{\delta x}p_h^S(w)\right]_{h=1}$$

That is, since $b^S_{k+1|k}(x|w)$ is the PHD of the spawning multitarget RFS $\Xi^S_{k+1|k}(w)$ with PGFl $p_h^S(w)$:

$$\begin{aligned} \left[\frac{\delta \Phi_w}{\delta x}[h]\right]_{h=1} &= p_s(w) f^E_{k+1|k}(x|w) p_1^S(w) + (1 - p_s(w) + p_s(w).p_1^E(w)) b^S_{k+1|k}(x|w) && (\textit{using } (23)) \\ &= p_s(w) f^E_{k+1|k}(x|w).1 + (1 - p_s(w) + p_s(w).1) b^S_{k+1|k}(x|w) \\ &= p_s(w) f^E_{k+1|k}(x|w) + b^S_{k+1|k}(x|w) \end{aligned}$$

Hence, $D = \int \left(p_s(w) f^E_{k+1|k}(x|w) + b^S_{k+1|k}(x|w)\right) D_{k|k}(w)dw$ and finally:

$$D_{k+1|k}(x) = A.B + C.D = b^B_{k+1|k}(x) + \int \left(p_s(w) f^E_{k+1|k}(x|w) + b^S_{k+1|k}(x|w)\right) D_{k|k}(w)dw$$

□

8.10 Proposition 4.2

Proof. Since f_Ξ is Poisson, $f_\Xi(X) = e^{-\lambda} \prod_{x \in X} I(x)$. Let $X = \{x_1, .., x_n\}$ be a collection of distinct elements from $\mathcal{X}$. Then:

$$\begin{aligned}
D_\Xi(X) &= \int f_\Xi(\{x_1, ..., x_n\} \cup W)\delta W \\
&= \sum_{p=0}^{\infty} \frac{1}{p!} \int_{\mathcal{X}^p} j_{\Xi, p+n}(x_1, ..., x_n, w_1, ..., w_p) dw_1 ... dw_p \\
&= \sum_{p=0}^{\infty} \frac{1}{p!} \int_{\mathcal{X}^p} e^{-\lambda} \prod_{i=1}^{n} (I(x_i)) \,.\, \prod_{j=1}^{p} (I(w_j))\, dw_1 ... dw_p \\
&= \prod_{i=1}^{n} (I(x_i))\,.e^{-\lambda}. \sum_{p=0}^{\infty} \frac{1}{p!} \int_{\mathcal{X}^p} \prod_{j=1}^{p} (I(w_j))\, dw_1 ... dw_p \\
&= \prod_{i=1}^{n} (I(x_i))\,.e^{-\lambda}. \sum_{p=0}^{\infty} \frac{1}{p!} \left(\int_{\mathcal{X}} I(w)dw\right)^p \\
&= \prod_{i=1}^{n} (I(x_i))\,.e^{-\lambda}. \sum_{p=0}^{\infty} \frac{1}{p!} \lambda^p \\
&= \prod_{i=1}^{n} I(x_i)
\end{aligned}$$

Furthermore:

$$\begin{aligned}
G_\Xi[h] &= \int h^W f_\Xi(W)\delta W \\
&= \sum_{p=0}^{\infty} \frac{1}{p!} \int_{\mathcal{X}^p} \prod_{i=1}^{p} (h(w_i))\, j_{\Xi,p}(w_1, ..., w_p) dw_1 ... dw_p \\
&= \sum_{p=0}^{\infty} \frac{1}{p!} \int_{\mathcal{X}^p} e^{-\lambda} \prod_{i=1}^{p} (h(w_i)) \prod_{i=1}^{p} (I(w_i))\, dw_1 ... dw_p \\
&= e^{-\lambda}. \sum_{p=0}^{\infty} \frac{1}{p!} \left(\int_{\mathcal{X}} h(w) I(w)dw\right)^p \\
&= e^{-\lambda}. \exp\left(\int_{\mathcal{X}} h(w) I(w)dw\right) \\
&= e^{I[h]-\lambda}
\end{aligned}$$

□

8.11 Proposition 4.3

Proof. Conditionnaly on the multi-target set $X = \{x_k^1, ..., x_k^n\}$ the single-sensor/multi-target observation RFS Σ_{k+1} equals to:

$$\Sigma_{k+1} = \Sigma_{k+1}^O(x_k^1) \cup ... \cup \Sigma_{k+1}^O(x_k^n) \cup \Sigma_{k+1}^{FA}$$

where:

- $\Sigma_{k+1}^O(x_k^i)$ is the RFS describing the observation of the i-th existing target;
- Σ_{k+1}^{FA} is the RFS describing the false alarm process.

Since $\Sigma_{k+1}^O(x_k^i)$ equals to:

- $\emptyset$ with probability $1 - p_d(x_k^i)$ (target i undetected);
- $\{z\}, z \in \mathcal{Z}$ with probability $f_{k+1}^O(z|x_k^i)p_d(x_k^i)$ (target i produces a measurement z);
- $Z \in \mathcal{Z}^n, n \geqslant 2$ with zero probability.

its PGFl $G_{\Sigma_{k+1}^O(x_k^i)}$ equals to:

$$\begin{aligned}
G_{\Sigma_{k+1}^O(x_k^i)}[g] &= \int g^Z f_{k+1|k}^O(Z|x_k^i)\delta Z \\
&= \underbrace{j_{\Sigma_{k+1}^O(x_k^i),0}(\emptyset)}_{=1-p_d(x_k^i)} + \int_{\mathcal{Z}^1} g(z_1) \underbrace{j_{\Sigma_{k+1}^O(x_k^i),1}(z_1)}_{=f_{k+1}^O(z|x_k^i)p_d(x_k^i)} dy_1 \\
&\quad + \sum_{p=2}^{\infty} \frac{1}{p!}\int_{\mathcal{Z}^p} g(z_1)...g(z_p) \underbrace{j_{\Sigma_{k+1}^O(x_k^i),p}(z_1, ..., z_p)}_{=0} dz_1...dz_p \\
&= 1 - p_d(x_k^i) + \int g(z) f_{k+1}^O(z|x_k^i)p_d(x_k^i)dz \\
&= 1 - p_d(_k^i) + p_d(x_k^i)p_g^O(x_k^i)
\end{aligned}$$

Since the RFS Σ_{k+1}^{FA} is Poisson, according to proposition 4.2 its PGFl $G_{\Sigma_{k+1}^{FA}}$ equals to $G_{\Sigma_{k+1}^{FA}}[g] = e^{\lambda.c[g]-\lambda}$. Then, using the product rule of independent PGFls (7), we have:

$$\begin{aligned}
G_{k+1}[g|X] &= \prod_{x\in X} \left(G_{\Sigma_{k+1}^O(x)}[g]\right).G_{\Sigma_{k+1}^{FA}}[g] \\
&= (1 - p_d + p_d.p_g^O)^X.e^{\lambda.c[g]-\lambda}
\end{aligned}$$

□

8.12 Proposition 4.4

Proof.

We have:

$$\begin{aligned}
\beta[g,\delta_x] &= \frac{\delta}{\delta x}\beta[g,h] \\
&= \frac{\delta}{\delta x}\left(\lambda c[g] - \lambda + \mu s[h(1-p_d)] + \mu s[hp_d.p_g^O] - \mu\right) \\
&= \mu\frac{\delta}{\delta x}\left(\int h(w)(1-p_d(w))s(w)dw\right) + \mu\frac{\delta}{\delta x}\left(\int h(w)p_d(w)p_g^O(w)s(w)dw\right) \\
&= \mu(1-p_d(x))s(x) + \mu p_d(x)p_g^O(x)s(x) && (using\ (14))
\end{aligned}$$

Furthermore:

$$\begin{aligned}
\beta[\delta_z,h] &= \frac{\delta}{\delta z}\beta[g,h] \\
&= \frac{\delta}{\delta z}\left(\lambda c[g] - \lambda + \mu s[h(1-p_d)] + \mu s[hp_d.p_g^O] - \mu\right) \\
&= \lambda\frac{\delta}{\delta z}\left(\int g(u)c(u)du\right) + \mu\int h(x)p_d(x)\frac{\delta}{\delta z}\left(\int g(u)f_{k+1}(u|x)du\right)s(x)dx \\
&= \lambda c(z) + \mu\int h(x)p_d(x)\underbrace{f_{k+1}(z|x)}_{=L_z(x)}s(x)dx && (using\ (14)) \\
&= \lambda c(z) + \mu s[hp_dL_z]
\end{aligned}$$

Finally:

$$\begin{aligned}
\beta[\delta_z,\delta_x] &= \frac{\delta}{\delta x}\beta[\delta_z,h] \\
&= \frac{\delta}{\delta x}\left(\lambda c(z) + \mu s[hp_dL_z]\right) \\
&= \mu\frac{\delta}{\delta x}\left(\int h(w)p_d(w)L_z(w)s(w)dw\right) \\
&= \mu p_d(x)L_z(x)s(x) && (using\ (14))
\end{aligned}$$

□

8.13 Theorem 4.2

Proof. Let $F[g,h]$ be the two-variable PGFl defined by:

$$\begin{aligned}
F[g,h] &= \iint h^X g^Z f_{k+1}(Z|X)f_{k+1|k}(X|Z^{(k)})\delta X\delta Z \\
&= \int h^X\left(\int g^Z f_{k+1}(Z|X)\delta Z\right)f_{k+1|k}(X|Z^{(k)})\delta X \\
&= \int h^X G_{k+1}[g|X]f_{k+1|k}(X|Z^{(k)})\delta X
\end{aligned}$$

Let $Z_{k+1} = \{z_1, ..., z_m\}$ be the collection of current measurements. Differentiating $F[g,h]$ in points $z_1, ..., z_m$ gives:

$$\begin{aligned}
\left[\frac{\delta^m}{\delta z_1 ... \delta z_m} F[g,h]\right]_{g=0} &= \int h^X \left[\frac{\delta^m}{\delta z_1 ... \delta z_m} G_{k+1}[g|X]\right]_{g=0} f_{k+1|k}(X|Z^{(k)})\delta X \\
&= \int h^X f_{k+1}(Z_{k+1}|X) f_{k+1|k}(X|Z^{(k)})\delta X \qquad (using (15))
\end{aligned}$$

Therefore, the PGFl $G_{k+1|k+1}$ of data update multi-target RFS $\Xi_{k+1|k+1}$ is:

$$\begin{aligned}
G_{k+1|k+1}[h] &= \int h^X f_{k+1|k+1}(X|Z^{(k+1)})\delta X \qquad (using (5)) \\
&= \int h^X \left(\frac{f_{k+1}(Z_{k+1}|X) f_{k+1|k}(X|Z^{(k)})}{\int f_{k+1}(Z_{k+1}|W) f_{k+1|k}(W|Z^{(k)})\delta W}\right)\delta X \qquad (using (2)) \\
&= \frac{\int h^X f_{k+1}(Z_{k+1}|X) f_{k+1|k}(X|Z^{(k)})\delta X}{\int f_{k+1}(Z_{k+1}|W) f_{k+1|k}(W|Z^{(k)})\delta W} \\
&= \frac{\left[\frac{\delta^m}{\delta z_1 ... \delta z_n} F[g,h]\right]_{g=0}}{\left[\frac{\delta^n}{\delta z_1 ... \delta z_m} F[g,h]\right]_{g=0,h=1}}
\end{aligned}$$

Then, using (23), we have:

$$\begin{aligned}
D_{k+1|k+1}(x) &= \left[\frac{\delta}{\delta x} G_{k+1|k+1}[h]\right]_{h=1} \\
&= \frac{\left[\frac{\delta^{n+1}}{\delta z_1 ... \delta z_m \delta x} F[g,h]\right]_{g=0,h=1}}{\left[\frac{\delta^m}{\delta z_1 ... \delta z_m} F[g,h]\right]_{g=0,h=1}}
\end{aligned}$$

That is, the data updated PHD $D_{k+1|k+1}$ is the ratio of set derivatives of the fundamental PGFl $F[g,h]$. Now, $F[g,h]$ must be constructed explicitely. We have:

$$\begin{aligned}
F[g,h] &= \int h^X G_{k+1}[g|X] f_{k+1|k}(X|Z^{(k)})\delta X \\
&= \int h^X (1 - p_d + p_d.p_g^O)^X . e^{\lambda c[g] - \lambda} f_{k+1|k}(X|Z^{(k)})\delta X \qquad (using\ (31)) \\
&= e^{\lambda c[g] - \lambda} . \int \left(h(1 - p_d + p_d.p_g^O)\right)^X f_{k+1|k}(X|Z^{(k)})\delta X \\
&= e^{\lambda c[g] - \lambda} . G_{k+1|k}[h(1 - p_d + p_d.p_g^O)]
\end{aligned}$$

Since $f_{k+1|k}(X|Z^{(k)})$ is assumed approximately Poisson with intensity $\mu s(x)$ and parameter μ, according to proposition 4.2 we have $G_{k+1|k}[h] \simeq e^{\mu.s[h] - \mu}$ with $s[h] = \int h(x)s(x)dx$. Therefore:

$$\begin{aligned}
F[g,h] &= e^{\lambda c[g] - \lambda} . G_{k+1|k}[h(1 - p_d + p_d.p_g^O)] \\
&\simeq e^{\lambda c[g] - \lambda + \mu s[h(1 - p_d + p_d.p_g^O)] - \mu} \\
&\simeq e^{\beta[g,h]}
\end{aligned}$$

The data updated PHD $D_{k+1|k+1}$ can therefore be computed using set derivatives of the generic cross-term *only*:

$$\begin{aligned}
D_{k+1|k+1}(x) &= \frac{\left[\frac{\delta^{m+1}}{\delta z_1 \dots \delta z_n \delta x} F[g,h]\right]_{g=0,h=1}}{\left[\frac{\delta^{m}}{\delta z_1 \dots \delta z_m} F[g,h]\right]_{g=0,h=1}} \\
&\simeq \frac{\left[\frac{\delta}{\delta x}\left(\frac{\delta^{m}}{\delta z_1 \dots \delta z_m} e^{\beta[g,h]}\right)\right]_{g=0,h=1}}{\left[\frac{\delta^{m}}{\delta z_1 \dots \delta z_m} e^{\beta[g,h]}\right]_{g=0,h=1}} \\
&\simeq \frac{\left[\frac{\delta}{\delta x}\left(e^{\beta[g,h]} \prod_{i=1}^{m} \beta[\delta_{z_i}, h]\right)\right]_{g=0,h=1}}{\left[e^{\beta[g,h]} \prod_{i=1}^{m} \beta[\delta_{z_i}, h]\right]_{g=0,h=1}} \\
&\simeq \frac{\left[\beta[g,\delta_x].e^{\beta[g,h]}.\prod_{i=1}^{m} \beta[\delta_{z_i}, h] + e^{\beta[g,h]}.\sum_{i=1}^{m}\left(\beta[\delta_{z_i},\delta_x].\prod_{j\neq i}\beta[\delta_{z_j},h]\right)\right]_{g=0,h=1}}{\left[e^{\beta[g,h]} \prod_{i=1}^{m} \beta[\delta_{z_i}, h]\right]_{g=0,h=1}} \\
&\simeq \beta[0,\delta_x] + \sum_{z\in Z_{k+1}} \frac{\beta[\delta_z,\delta_x]}{\beta[\delta_z,1]} \\
&\simeq \mu(1-p_d(x))s(x) + \sum_{z\in Z_{k+1}} \frac{\mu p_d(x) L_z(x) s(x)}{\lambda c(z) + \mu s[p_d L_z]} \\
&\simeq \left(1 - p_d(x) + \sum_{z\in Z_{k+1}} \frac{p_d(x) L_z(x)}{\lambda c(z) + \mu s[p_d L_z]}\right) \mu s(x)
\end{aligned}$$

Finally, since the time updated RFS $\Sigma_{k+1|k}$ is assumed Poisson, its intensity $\mu s(x)$ equals its first order moment $D_{k+1|k}(\{x\})$ (proposition 4.2) or, equivalently, its PHD $D_{k+1|k}(x)$ (equation (22)):

$$D_{k+1|k+1}(x) \simeq \left(1 - p_d(x) + \sum_{z\in Z_{k+1}} \frac{p_d(x) L_z(x)}{\lambda c(z) + D_{k+1|k}[p_d L_z]}\right) D_{k+1|k}(x)$$

$\square$

8.14 Proposition 5.1

Proof.

We have:

$$
\begin{aligned}
&\beta[g^{[1]}, ..., g^{[N]}, \delta_x] \\
&= \frac{\delta}{\delta x}\beta[g^{[1]}, ..., g^{[N]}, h] \\
&= \frac{\delta}{\delta x}\left(\sum_{j=1}^{N}(\lambda^{[j]}c^{[j]}[g^{[j]}] - \lambda^{[j]}) + \mu s\left[h\prod_{j=1}^{N}\left(1 - p_d^{[j]} + p_d^{[j]}p_{g^{[j]}}^{O,[j]}\right)\right] - \mu\right) \\
&= \mu\frac{\delta}{\delta x}\left(\int h(w)\prod_{j=1}^{N}\left(1 - p_d^{[j]}(w) + p_d^{[j]}(w)p_{g^{[j]}}^{O,[j]}(w)\right)s(w)dw\right) \\
&= \mu\prod_{j=1}^{N}\left(1 - p_d^{[j]}(x) + p_d^{[j]}(x)p_{g^{[j]}}^{O,[j]}(x)\right)s(x)
\end{aligned}
$$

Furthermore:

$$
\begin{aligned}
&\beta[g^{[1]}, ..., \delta_{z^{[t_1]}}, ..., \delta_{z^{[t_M]}}, ..., g^{[N]}, h] \\
&= \frac{\delta^M}{\delta z^{[t_1]}...\delta z^{[t_M]}}\beta[g^{[1]}, ..., g^{[N]}, h] \\
&= \frac{\delta^M}{\delta z^{[t_1]}...\delta z^{[t_M]}}\left(\sum_{j=1}^{N}(\lambda^{[j]}c^{[j]}[g^{[j]}] - \lambda^{[j]}) + \mu s\left[h\prod_{j=1}^{N}\left(1 - p_d^{[j]} + p_d^{[j]}p_{g^{[j]}}^{O,[j]}\right)\right] - \mu\right) \\
&= \begin{cases}
\lambda^{[t_1]}c^{[t_1]}(z^{[t_1]}) + \mu\int h(x)p_d^{[t_1]}(x)\dfrac{\delta\left(\int g^{[t_1]}(z)f_{k+1}^{0,[t_1]}(z|x)dz\right)}{\delta z^{[t_1]}}\prod_{j\neq t_1}\left(1 - p_d^{[j]}(x) + p_d^{[j]}(x)p_{g^{[j]}}^{O,[j]}(x)\right)s(x)dx & (M=1) \\
\mu\int h(x)\prod_{i=1}^{M}\left(p_d^{[t_i]}(x)\dfrac{\delta\left(\int g^{[t_i]}(z)f_{k+1}^{O,[t_i]}(z|x)dz\right)}{\delta z^{[t_i]}}\right)\prod_{j\notin\{t_1,...,t_M\}}\left(1 - p_d^{[j]}(x) + p_d^{[j]}(x)p_{g^{[j]}}^{O,[j]}(x)\right)s(x)dx & (M>1)
\end{cases} \\
&= \begin{cases}
\lambda^{[t_1]}c^{[t_1]}(z^{[t_1]}) + \mu\int h(x)p_d^{[t_1]}(x)\underbrace{f_{k+1}^{O,[t_1]}(z^{[t_1]}|x)}_{=L_{z^{[t_1]}}^{[t_1]}(x)}\prod_{j\neq t_1}\left(1 - p_d^{[j]}(x) + p_d^{[j]}(x)p_{g^{[j]}}^{O,[j]}(x)\right)s(x)dx & (M=1) \\
\mu\int h(x)\prod_{i=1}^{M}\left(p_d^{[t_i]}(x)\underbrace{f_{k+1}^{O,[t_i]}(z^{[t_i]}|x)}_{=L_{z^{[t_i]}}^{[t_i]}(x)}\right)\prod_{j\notin\{t_1,...,t_M\}}\left(1 - p_d^{[j]}(x) + p_d^{[j]}(x)p_{g^{[j]}}^{O,[j]}(x)\right)s(x)dx & (M>1)
\end{cases} \\
&= \begin{cases}
\lambda^{[t_1]}c^{[t_1]}(z^{[t_1]}) + \mu s\left[hp_d^{[t_1]}L_{z^{[t_1]}}^{[t_1]}\prod_{j\neq\{t_1,...,t_M\}}\left(1 - p_d^{[j]} + p_d^{[j]}p_{g^{[j]}}^{O,[j]}\right)\right] & (M=1) \\
\mu s\left[h\prod_{i=1}^{M}\left(p_d^{[t_i]}L_{z^{[t_i]}}^{[t_i]}\right)\prod_{j\notin\{t_1,...,t_M\}}\left(1 - p_d^{[j]} + p_d^{[j]}p_{g^{[j]}}^{O,[j]}\right)\right] & (M>1)
\end{cases}
\end{aligned}
$$

Finally:

$$
\begin{aligned}
&\beta[g^{[1]}, ..., \delta_{z^{[t_1]}}, ..., \delta_{z^{[t_M]}}, ..., g^{[N]}, \delta_x] \\
&= \frac{\delta}{\delta x}\beta[g^{[1]}, ..., \delta_{z^{[t_1]}}, ..., \delta_{z^{[t_M]}}, ..., g^{[N]}, h] \\
&= \begin{cases}
\mu \dfrac{\delta}{\delta x}\left(\displaystyle\int h(w) p_d^{[t_1]}(w) L_{z^{[t_1]}}^{[t_1]}(w) \prod_{j \neq t_1} \left(1 - p_d^{[j]}(w) + p_d^{[j]}(w) p_{g^{[j]}}^{O,[j]}(w)\right) s(w) dw\right) & (M = 1) \\
\mu \dfrac{\delta}{\delta x}\left(\displaystyle\int h(w) \prod_{i=1}^{M} \left(p_d^{[t_i]}(w) L_{z^{[t_i]}}^{[t_i]}(w)\right) \prod_{j \notin \{t_1, ..., t_M\}} \left(1 - p_d^{[j]}(w) + p_d^{[j]}(w) p_{g^{[j]}}^{O,[j]}(w)\right) s(w) dw\right) & (M > 1)
\end{cases} \\
&= \mu \prod_{i=1}^{M} \left(p_d^{[t_i]}(x) L_{z^{[t_i]}}^{[t_i]}(x)\right) \prod_{j \notin \{t_1, ..., t_M\}} \left(1 - p_d^{[j]}(x) + p_d^{[j]}(x) p_{g^{[j]}}^{O,[j]}(x)\right) s(x) \qquad (u
\end{aligned}
$$

□

8.15 Proposition 5.2

Proof. Let $\Sigma_{k+1}^{[j]}$ be the observation RFS of the j-th sensor, with density $f_{k+1}^{[j]}$ and PGFl $G_{k+1}^{[j]}$. The current observation Z_{k+1} is the "union" of current observations $Z_{k+1}^{[j]}$ which are defined on (eventually) different single-observation space $\mathcal{Z}^{[j]}$. Therefore, the set-based data update equation (2) can be written as follows:

$$
\begin{aligned}
f_{k+1|k+1}(X|Z^{(k+1)}) &= \frac{f_{k+1}(Z_{k+1}|X) f_{k+1|k}(X|Z^{(k)})}{\int f_{k+1}(Z_{k+1}|W) f_{k+1|k}(W|Z^{(k)}) \delta W} \\
&= \frac{\prod_{j=1}^{N} \left(f_{k+1}^{[j]}(Z_{k+1}^{[j]}|X)\right) f_{k+1|k}(X|Z^{(k)})}{\int \prod_{j=1}^{N} \left(f_{k+1}^{[j]}(Z_{k+1}^{[j]}|W)\right) f_{k+1|k}(W|Z^{(k)}) \delta W}
\end{aligned}
$$

Let $F[g^{[1]}, ..., g^{[N]}, h]$ be the $N+1$-variable PGFl defined as:

$$
\begin{aligned}
F[g^{[1]}, ..., g^{[N]}, h] &= \int \cdots \int h^X \prod_{j=1}^{N} \left((g^{[j]})^{Z^{[j]}} f_{k+1}^{[j]}(Z^{[j]}|X)\right) \delta X \delta Z^{[1]} ... \delta Z^{[N]} \\
&= \int h^X \prod_{j=1}^{N} \left(\int (g^{[j]})^{Z^{[j]}} f_{k+1}^{[j]}(Z^{[j]}|X) \delta Z^{[j]}\right) f_{k+1|k}(X|Z^{(k)}) \delta X \\
&= \int h^X \prod_{j=1}^{N} \left(G_{k+1}^{[j]}[g^{[j]}|X]\right) f_{k+1|k}(X|Z^{(k)}) \delta X
\end{aligned}
$$

Let $Z^{[j]}_{k+1} = \{z^{[j]}_1, ..., z^{[j]}_{m[j]}\}$ be the collection of current measurements produced by sensor j. Differentiating $F[g,h]$ in points $z^{[1]}_1, ..., z^{[1]}_{m[1]}, ..., z^{[N]}_1, ..., z^{[N]}_{m[N]}$ gives:

$$\begin{aligned}
&\left[\frac{\delta^{m[1]+...+m[N]}}{\delta z^{[1]}_1 ... \delta z^{[1]}_{m[1]} ... \delta z^{[N]}_1 ... \delta z^{[N]}_{m[N]}} F[g^{[1]}, ..., g^{[N]}, h]\right]_{g^{[1]}=0,...,g^{[N]}=0} \\
&= \int h^X \prod_{j=1}^N \left(\left[\frac{\delta^{m[j]}}{\delta z^{[j]}_1 ... \delta z^{[j]}_{m[j]}} G^{[j]}_{k+1}[g^{[j]}|X]\right]_{g^{[j]}=0}\right) f_{k+1|k}(X|Z^{(k)})\delta X \\
&= \int h^X \prod_{j=1}^N \left(f^{[j]}_{k+1}(Z^{[j]}_{k+1}|X)\right) f_{k+1|k}(X|Z^{(k)})\delta X
\end{aligned}$$

Therefore, the PGFl $G_{k+1|k+1}$ of the data updated multi-target RFS $\Xi_{k+1|k+1}$ is:

$$\begin{aligned}
G_{k+1|k+1}[h] &= \int h^X f_{k+1|k+1}(X|Z^{(k+1)})\delta X \\
&= \int h^X \frac{\prod_{j=1}^N \left(f^{[j]}_{k+1}(Z^{[j]}_{k+1}|X)\right) f_{k+1|k}(X|Z^{(k)})}{\int \prod_{j=1}^N \left(f^{[j]}_{k+1}(Z^{[j]}_{k+1}|W)\right) f_{k+1|k}(W|Z^{(k)})\delta W}\delta X \\
&= \frac{\int h^X \prod_{j=1}^N \left(f^{[j]}_{k+1}(Z^{[j]}_{k+1}|X)\right) f_{k+1|k}(X|Z^{(k)})\delta X}{\int \prod_{j=1}^N \left(f^{[j]}_{k+1}(Z^{[j]}_{k+1}|W)\right) f_{k+1|k}(W|Z^{(k)})\delta W} \\
&= \frac{\left[\frac{\delta^{m[1]+...+m[N]}}{\delta z^{[1]}_1 ... \delta z^{[1]}_{m[1]} ... \delta z^{[N]}_1 ... \delta z^{[N]}_{m[N]}} F[g^{[1]}, ..., g^{[N]}, h]\right]_{g^{[1]}=0,...,g^{[N]}=0}}{\left[\frac{\delta^{m[1]+...+m[N]}}{\delta z^{[1]}_1 ... \delta z^{[1]}_{m[1]} ... \delta z^{[N]}_1 ... \delta z^{[N]}_{m[N]}} F[g^{[1]}, ..., g^{[N]}, h]\right]_{g^{[1]}=0,...,g^{[N]}=0,h=1}}
\end{aligned}$$

Then, using (23), we have:

$$\begin{aligned}
D_{k+1|k+1}(x) &= \left[\frac{\delta}{\delta x} G_{k+1|k+1}[h]\right]_{h=1} \\
&= \frac{\left[\frac{\delta}{\delta x}\left(\frac{\delta^{m[1]+...+m[N]}}{\delta z^{[1]}_1 ... \delta z^{[1]}_{m[1]} ... \delta z^{[N]}_1 ... \delta z^{[N]}_{m[N]}} F[g^{[1]}, ..., g^{[N]}, h]\right)\right]_{g^{[1]}=0,...,g^{[N]}=0,h=1}}{\left[\frac{\delta^{m[1]+...+m[N]}}{\delta z^{[1]}_1 ... \delta z^{[1]}_{m[1]} ... \delta z^{[N]}_1 ... \delta z^{[N]}_{m[N]}} F[g^{[1]}, ..., g^{[N]}, h]\right]_{g^{[1]}=0,...,g^{[N]}=0,h=1}}
\end{aligned}$$

That is, the data updated PHD $D_{k+1|k+1}$ is the ratio of set derivatives of the fundamental PGFl $F[g^{[1]},...,g^{[N]},h]$. This PGFl can be written:

$$\begin{aligned}
F[g^{[1]},...,g^{[N]},h] &= \int h^X \prod_{j=1}^{N}\left(G^{[j]}_{k+1}[g^{[j]}|X]\right) f_{k+1|k}(X|Z^{(k)})\delta X \\
&= \int h^X \prod_{j=1}^{N}\left(\left(1-p_d^{[j]}+p_d^{[j]}.p^{O,[j]}_{g^{[j]}}\right)^X e^{\lambda^{[j]}c^{[j]}[g^{[j]}]-\lambda^{[j]}}\right) f_{k+1|k}(X|Z^{(k)})\delta X \qquad (using\ (31)) \\
&= \prod_{j=1}^{N}\left(e^{\lambda^{[j]}c^{[j]}[g^{[j]}]-\lambda^{[j]}}\right).\int\left(h\prod_{j=1}^{N}\left(1-p_d^{[j]}+p_d^{[j]}.p^{O,[j]}_{g^{[j]}}\right)\right)^X f_{k+1|k}(X|Z^{(k)})\delta X \\
&= e^{\sum_{j=1}^{N}\left(\lambda^{[j]}c^{[j]}[g^{[j]}]-\lambda^{[j]}\right)}.G_{k+1|k}\left[h\prod_{j=1}^{N}\left(1-p_d^{[j]}+p_d^{[j]}.p^{O,[j]}_{g^{[j]}}\right)\right]
\end{aligned}$$

Since $f_{k+1|k}(X|Z^{(k)})$ is assumed approximately Poisson with parameter μ and intensity $\mu s(x)$, according to proposition 4.2 we have $G_{k+1|k}[h] \simeq e^{\mu s[h]-\mu}$ with $s[h]=\int h(x)s(x)dx$. Therefore:

$$\begin{aligned}
F[g,h] &= e^{\sum_{j=1}^{N}\left(\lambda^{[j]}c^{[j]}[g^{[j]}]-\lambda^{[j]}\right)}.G_{k+1|k}\left[h\prod_{j=1}^{N}\left(1-p_d^{[j]}+p_d^{[j]}.p^{O,[j]}_{g^{[j]}}\right)\right] \\
&\simeq e^{\sum_{j=1}^{N}\left(\lambda^{[j]}c^{[j]}[g^{[j]}]-\lambda^{[j]}\right)+\mu s\left[h\prod_{j=1}^{N}\left(1-p_d^{[j]}+p_d^{[j]}.p^{O,[j]}_{g^{[j]}}\right)\right]-\mu} \\
&\simeq e^{\beta[g^{[1]},...,g^{[N]},h]}
\end{aligned}$$

And finally:

$$\begin{aligned}
D_{k+1|k+1}(x) &= \frac{\left[\frac{\delta}{\delta x}\left(\frac{\delta^{m[1]+...+m[N]}}{\delta z_1^{[1]}...\delta z_{m[1]}^{[1]}...\delta z_1^{[N]}...\delta z_{m[N]}^{[N]}}F[g^{[1]},...,g^{[N]},h]\right)\right]_{g^{[1]}=0,...,g^{[N]}=0,h=1}}{\left[\frac{\delta^{m[1]+...+m[N]}}{\delta z_1^{[1]}...\delta z_{m[1]}^{[1]}...\delta z_1^{[N]}...\delta z_{m[N]}^{[N]}}F[g^{[1]},...,g^{[N]},h]\right]_{g^{[1]}=0,...,g^{[N]}=0,h=1}} \\
&\simeq \frac{\left[\frac{\delta}{\delta x}\left(\frac{\delta^{m[1]+...+m[N]}}{\delta z_1^{[1]}...\delta z_{m[1]}^{[1]}...\delta z_1^{[N]}...\delta z_{m[N]}^{[N]}}e^{\beta[g^{[1]},...,g^{[N]},h]}\right)\right]_{g^{[1]}=0,...,g^{[N]}=0,h=1}}{\left[\frac{\delta^{m[1]+...+m[N]}}{\delta z_1^{[1]}...\delta z_{m[1]}^{[1]}...\delta z_1^{[N]}...\delta z_{m[N]}^{[N]}}e^{\beta[g^{[1]},...,g^{[N]},h]}\right]_{g^{[1]}=0,...,g^{[N]}=0,h=1}}
\end{aligned}$$

□

8.16 Lemma 5.1

Proof. First, $\check{Z}_{M+1}$ can be constructed using $\check{Z}_M$ as follows:

$$
\begin{aligned}
\check{Z}_{M+1} &= \bigcup_{p=1}^{M+1} \check{Z}^p_{M+1} \\
&= \left(\bigcup_{p=1}^{M} \check{Z}^p_{M+1} \right) \cup \left(\check{Z}^{M+1}_{M+1} \right) \\
&= \left(\bigcup_{p=1}^{M} \bigcup_{\substack{I\in 1[,M+1] \\ |I|=p}} \prod_{i\in I} Z^{[i]} \right) \cup \left(Z^{[M+1]} \times \check{Z}^M_M \right) \\
&= \left(\bigcup_{p=1}^{M} \underbrace{\bigcup_{\substack{I\in [1\ M] \\ |I|=p}} \prod_{i\in I} Z^{[i]}}_{=\check{Z}^p_M} \right) \cup \left(\bigcup_{p=1}^{M} Z^{[M+1]} \times \underbrace{\bigcup_{\substack{I\in [1\ M] \\ |I|=p-1}} \prod_{i\in I} Z^{[i]}}_{=\emptyset if p=0} \right) \cup \left(Z^{[M+1]} \times \check{Z}^M_M \right) \\
&= \left(\bigcup_{p=1}^{M} \check{Z}^p_M \right) \cup Z^{[M+1]} \cup \left(Z^{[M+1]} \times \bigcup_{p=1}^{M-1} \underbrace{\bigcup_{\substack{I\in [1\ M] \\ |I|=p}} \prod_{i\in I} Z^{[i]}}_{=\check{Z}^p_M} \right) \cup \left(Z^{[M+1]} \times \check{Z}^M_M \right) \\
&= \check{Z}_M \cup Z^{[M+1]} \cup \left(Z^{[M+1]} \times \bigcup_{p=1}^{M-1} \check{Z}^p_M \right) \cup \left(Z^{[M+1]} \times \check{Z}^M_M \right) \\
&= \check{Z}_M \cup Z^{[M+1]} \cup \left(Z^{[M+1]} \times \bigcup_{p=1}^{M} \check{Z}^p_M \right) \\
&= \check{Z}_M \cup Z^{[M+1]} \cup \left(Z^{[M+1]} \times \check{Z}_M \right)
\end{aligned}
$$

We can therefore write:

$$
\begin{aligned}
&\mathcal{T}(\check{Z}_{M+1}) \\
&= \left\{ T \subseteq \check{Z}_{M+1} \mid \sup_{M+1}(T) = \prod_{z \in Z_{M+1}} (z,1) \right\} \\
&= \left\{ T^a \cup T^b \cup T^c \mid T^a \subseteq \check{Z}_M, T^b \subseteq Z^{[M+1]}, T^c \subseteq Z^{[M+1]} \times \check{Z}_M, \sup_{M+1}(T^a \cup T^b \cup T^c) = \prod_{z \in Z_{M+1}} (z,1) \right\} \\
&= \bigcup_{p=0}^{m[M+1]} \bigcup_{\substack{I \subseteq [1\ m[M+1]] \\ |I|=p}} \left\{ T^a \cup \{z_i^{[M+1]}\}_{i \notin I} \cup T^c \mid T^a \subseteq \check{Z}_M, T^c \subseteq Z^{[M+1]} \times \check{Z}_M, \sup_{M+1}(T^a \cup T^c) = \prod_{z \in Z_M \cup \{z_i^{[M+1]}\}_{i \in I}} (z,1) \right\} \\
&= \bigcup_{p=0}^{m[M+1]} \bigcup_{\substack{I \subseteq [1\ m[M+1]] \\ |I|=p}} \left\{ T^a \cup \{z_i^{[M+1]}\}_{i \notin I} \cup \{z_i^{[M+1]}, T_i\}_{i \in I} \mid T^a, T \subseteq \check{Z}_M, |T| = p, \sup_M(T^a \cup T) = \prod_{z \in Z_M} (z,1) \right\} \\
&= \bigcup_{p=0}^{m[M+1]} \bigcup_{\substack{T \subseteq \check{Z}_M \\ |T| \geqslant p \\ \sup_M(T) = \prod_{z \in Z_M} (z,1)}} \bigcup_{\substack{I \subseteq [1\ m[M+1]] \\ J \subseteq [1\ |T|] \\ |I|=|J|=p}} \bigcup_{\sigma \in Bij(I,J)} \left(\{T_j\}_{j \notin J} \cup \{z_i^{[M+1]}\}_{i \notin I} \cup \{z_i^{[M+1]}, T_{\sigma(i)}\}_{i \in I} \right) \\
&= \bigcup_{p=0}^{m[M+1]} \bigcup_{\substack{T \in \mathcal{T}(\check{Z}_M) \\ |T| \geqslant p}} \bigcup_{\substack{I \subseteq [1\ m[M+1]] \\ J \subseteq [1\ |T|] \\ |I|=|J|=p}} \bigcup_{\sigma \in Bij(I,J)} \left(\{T_j\}_{j \notin J} \cup \{z_i^{[M+1]}\}_{i \notin I} \cup \{z_i^{[M+1]}, T_{\sigma(i)}\}_{i \in I} \right) \\
&= \bigcup_{T \in \mathcal{T}(\check{Z}_M)} \bigcup_{p=0}^{\min(|T|, m[M+1])} \bigcup_{\substack{I \subseteq [1\ m[M+1]] \\ J \subseteq [1\ |T|] \\ |I|=|J|=p}} \bigcup_{\sigma \in Bij(I,J)} \left(\{T_j\}_{j \notin J} \cup \{z_i^{[M+1]}\}_{i \notin I} \cup \{z_i^{[M+1]}, T_{\sigma(i)}\}_{i \in I} \right)
\end{aligned}
$$

□

8.17 Proposition 5.3

Proof. First, let us prove by induction on $1 \leqslant M \leqslant N$ that:

$$
\frac{\delta^{m[1]+\ldots+m[M]}}{\delta z_1^{[1]} \ldots \delta z_{m[1]}^{[1]} \ldots \delta z_1^{[M]} \ldots \delta z_{m[M]}^{[M]}} e^{\tilde{\beta}[\emptyset,h]} = e^{\tilde{\beta}[\emptyset,h]} \sum_{T \in \mathcal{T}(\check{Z}_M)} \prod_{T_i \in T} \tilde{\beta}[T_i, h]
$$

Considering the measurements produced by the first sensor only, there is two cases to consider : either the first sensor produced no measurements ($m[1] = 0$) or produced at least one measurement ($m[1] > 0$). If $m[1] = 0$, then $Z^{[1]} = \emptyset$ and $\check{Z}_1 = \emptyset$. Therefore, according to definition (45), the set of combinational terms $\mathcal{T}(\check{Z}_1)$ is empty. Thus we have:

$$\begin{aligned}\frac{\delta^0}{\delta\emptyset}e^{\tilde{\beta}[\emptyset,h]} &= e^{\tilde{\beta}[\emptyset,h]} \\ &= e^{\tilde{\beta}[\emptyset,h]}\sum_{T\in\emptyset}\prod_{T_i\in T}\tilde{\beta}[T_i,h]\end{aligned}$$

If $m[1]>0$, then $\check{Z}_1$ is the set of 1-tuples $\{(z)\}_{z\in Z^{[1]}}$ and the set of combinational terms is reduced to the single element $\mathcal{T}(\check{Z}_1)=\{\check{Z}_1\}$. Thus we have:

$$\begin{aligned}\frac{\delta^{m[1]}}{\delta z_1^{[1]}...\delta z_{m[1]}^{[1]}}e^{\tilde{\beta}[\emptyset,h]} &= e^{\tilde{\beta}[\emptyset,h]}\prod_{i=1}^{m[1]}\tilde{\beta}[\delta_{z_i^{[1]}},g^{[2]},...,g^{[N]},h] \\ &= e^{\tilde{\beta}[\emptyset,h]}\prod_{z\in Z^{[1]}}\tilde{\beta}[(z),h] \\ &= e^{\tilde{\beta}[\emptyset,h]}\sum_{T=Z^{[1]}}\prod_{T_i\in T}\tilde{\beta}[T_i,h] \\ &= e^{\tilde{\beta}[\emptyset,h]}\sum_{T\in\mathcal{T}(\check{Z}_1)}\prod_{T_i\in T}\tilde{\beta}[T_i,h]\end{aligned}$$

Therefore, the case holds for $M=1$. Assuming that the case holds for $M<N$, let us prove that it holds for $M+1$. We can write:

$$\begin{aligned}&\frac{\delta^{m[1]+...+m[M]+m[M+1]}}{\delta z_1^{[1]}...\delta z_{m[1]}^{[1]}...\delta z_1^{[M]}...\delta z_{m[M]}^{[M]}\delta z_1^{[M+1]}...\delta z_{m[M+1]}^{[M+1]}}e^{\tilde{\beta}[\emptyset,h]} \\ &= \frac{\delta^{m[M+1]}}{\delta z_1^{[M+1]}...\delta z_{m[M+1]}^{[M+1]}}\left(\frac{\delta^{m[1]+...+m[M]}}{\delta z_1^{[1]}...\delta z_{m[1]}^{[1]}...\delta z_1^{[M]}...\delta z_{m[M]}^{[M]}}e^{\tilde{\beta}[\emptyset,h]}\right) \\ &= \frac{\delta^{m[M+1]}}{\delta z_1^{[M+1]}...\delta z_{m[M+1]}^{[M+1]}}\left(e^{\tilde{\beta}[\emptyset,h]}\sum_{T\in\mathcal{T}(\check{Z}_M)}\prod_{T_k\in T}\tilde{\beta}[T_k,h]\right) \quad \textit{(using case at step M)} \\ &= \sum_{p=0}^{m[M+1]}\sum_{\substack{I\subseteq[1\ m[M+1]]\\|I|=p}}\left(\frac{\delta^{m[M+1]-p}}{\delta\{z_i^{[M+1]}\}_{i\notin I}}e^{\tilde{\beta}[\emptyset,h]}\frac{\delta^p}{\delta\{z_i^{[M+1]}\}_{i\in I}}\left(\sum_{T\in\mathcal{T}(\check{Z}_M)}\prod_{T_k\in T}\tilde{\beta}[T_k,h]\right)\right) \\ &= \sum_{p=0}^{m[M+1]}\sum_{\substack{I\subseteq[1\ m[M+1]]\\|I|=p}}\left(e^{\tilde{\beta}[\emptyset,h]}\left(\prod_{i\notin I}\tilde{\beta}[(z_i^{[M+1]}),h]\right)\sum_{T\in\mathcal{T}(\check{Z}_M)}\underbrace{\left(\frac{\delta^p}{\delta\{z_i^{[M+1]}\}_{i\in I}}\prod_{T_k\in T}\tilde{\beta}[T_k,h]\right)}_{=A}\right)\end{aligned}$$

Since any cross-term is null if set on two different measurement point from the same observation space (i.e. the same sensor), A can be expanded as follows:

$$A=\begin{cases}0 & (p>|T|) \\ \displaystyle\sum_{\substack{J\subseteq[1\ |T|]\\|J|=p}}\sum_{\sigma\in Bij(I,J)}\prod_{j\notin J}\tilde{\beta}[T_j,h]\prod_{i\in I}\tilde{\beta}[(z_i^{[M+1]},T_{\sigma(i)}),h] & (p\leqslant|T|)\end{cases}$$

And therefore:

$$\frac{\delta^{m[1]+\ldots+m[M]+m[M+1]}}{\delta z_1^{[1]}\ldots\delta z_{m[1]}^{[1]}\ldots\delta z_1^{[M]}\ldots\delta z_{m[M]}^{[M]}\delta z_1^{[M+1]}\ldots\delta z_{m[M+1]}^{[M+1]}} e^{\tilde{\beta}[\emptyset,h]}$$
$$= e^{\tilde{\beta}[\emptyset,h]} \sum_{p=0}^{m[M+1]} \sum_{\substack{I\subseteq[1\ m[M+1]]\\|I|=p}} \sum_{\substack{T\in\mathcal{T}(\check{Z}_M)\\|T|\geqslant p}} \sum_{\substack{J\subseteq[1\ |T|]\\|J|=p}} \sum_{\sigma\in Bij(I,J)} \left(\prod_{i\notin I}\tilde{\beta}[(z_i^{[M+1]}),h]\right)\left(\prod_{j\notin J}\tilde{\beta}[T_j,h]\right)\left(\prod_{i\in I}\tilde{\beta}[(z_i^{[M+1]},T_{\sigma(i)}),h]\right)$$
$$= e^{\tilde{\beta}[\emptyset,h]} \sum_{T\in\mathcal{T}(\check{Z}_M)} \sum_{p=0}^{\min(|T|,m[M+1])} \sum_{\substack{I\subseteq[1\ m[M+1]]\\J\subseteq[1\ |T|]\\|I|=|J|=p}} \sum_{\sigma\in Bij(I,J)} \left(\prod_{i\notin I}\tilde{\beta}[(z_i^{[M+1]}),h]\right)\left(\prod_{j\notin J}\tilde{\beta}[T_j,h]\right)\left(\prod_{i\in I}\tilde{\beta}[(z_i^{[M+1]},T_{\sigma(i)}),h]\right)$$
$$= e^{\tilde{\beta}[\emptyset,h]} \bigcup_{T\in\mathcal{T}(\check{Z}_{M+1})} \prod_{T_i\in T} \tilde{\beta}[T_i,h] \quad (\textit{using lemma } 5.1)$$

Therefore, the case holds true for each $M \leqslant N$ and we have:

$$\left[\frac{\delta^{m[1]+\ldots+m[N]}}{\delta z_1^{[1]}\ldots\delta z_{m[1]}^{[1]}\ldots\delta z_1^{[N]}\ldots\delta z_{m[N]}^{[N]}} e^{\beta[g^{[1]},\ldots,g^{[N]},h]}\right]_{g^{[1]}=0,\ldots,g^{[N]}=0} = e^{\beta[\emptyset,h]} \bigcup_{T\in\mathcal{T}(\check{Z}_{M+1})} \prod_{T_i\in T} \beta[T_i,h]$$

Using proposition 5.2, we can finally write for any $x \in \mathcal{X}$:

$$D_{k+1|k+1}(x)$$
$$= \frac{\left[\frac{\delta}{\delta x}\left(\frac{\delta^{m[1]+\ldots+m[N]}}{\delta z_1^{[1]}\ldots\delta z_{m[1]}^{[1]}\ldots\delta z_1^{[N]}\ldots\delta z_{m[N]}^{[N]}} e^{\beta[g^{[1]},\ldots,g^{[N]},h]}\right)\right]_{g^{[1]}=0,\ldots,g^{[N]}=0,h=1}}{\left[\frac{\delta^{m[1]+\ldots+m[N]}}{\delta z_1^{[1]}\ldots\delta z_{m[1]}^{[1]}\ldots\delta z_1^{[N]}\ldots\delta z_{m[N]}^{[N]}} e^{\beta[g^{[1]},\ldots,g^{[N]},h]}\right]_{g^{[1]}=0,\ldots,g^{[N]}=0,h=1}}$$
$$= \frac{\left[\frac{\delta}{\delta x}\left(e^{\beta[\emptyset,h]} \bigcup_{T\in\mathcal{T}(\check{Z}_{M+1})} \prod_{T_i\in T} \beta[T_i,h]\right)\right]_{h=1}}{\left[e^{\beta[\emptyset,h]} \bigcup_{T\in\mathcal{T}(\check{Z}_{M+1})} \prod_{T_i\in T} \beta[T_i,h]\right]_{h=1}}$$
$$= \beta[\emptyset,\delta_x] + \frac{\sum_{T\in\mathcal{T}(\check{Z}_N)} \sum_{T_i\in T} \left(\beta[T_i,\delta_x] \prod_{\substack{T_j\in T\\T_j\neq T_i}} \beta[T_j,1]\right)}{\sum_{T\in\mathcal{T}(\check{Z}_N)} \prod_{T_i\in T} \beta[T_i,1]}$$

□

8.18 Proposition 6.1

Proof. The multi-sensor cross-term set in target state x can be reformulated:

$$
\begin{aligned}
&\beta[g^{[1]},...,g^{[N]},\delta_x]\\
&=\mu\prod_{j=1}^{N}\left(1-p_d^{[j]}(x)+p_d^{[j]}(x)p_{g^{[j]}}^{O,[j]}(x)\right)s(x)\\
&=\mu\prod_{q=1}^{P}\left(\prod_{j\in P_s(q)}\left(1-p_d^{[j]}(x)+p_d^{[j]}(x)p_{g^{[j]}}^{O,[j]}(x)\right)\right)s(x)
\end{aligned}
$$

If $\exists p\in[1\ P], x\in P_t(p)$ then:

$$
\begin{aligned}
&\beta[g^{[1]},...,g^{[N]},\delta_x]\\
&=\mu\prod_{j\in P_s(p)}\left(1-p_d^{[j]}(x)+p_d^{[j]}(x)p_{g^{[j]}}^{O,[j]}(x)\right)\prod_{q\neq p}\left(\prod_{j\in P_s(q)}\left(1-\underbrace{p_d^{[j]}(x)}_{=0}+\underbrace{p_d^{[j]}(x)}_{=0}p_{g^{[j]}}^{O,[j]}(x)\right)\right)s(x)\\
&=\mu\prod_{j\in P_s(p)}\left(1-p_d^{[j]}(x)+p_d^{[j]}(x)p_{g^{[j]}}^{O,[j]}(x)\right)s(x)\\
&=\beta_p[g^{[1]},...,g^{[N]},\delta_x]
\end{aligned}
$$

Otherwise, $x\in P_t(0)$ and therefore:

$$
\begin{aligned}
&\beta[g^{[1]},...,g^{[N]},\delta_x]\\
&=\mu\prod_{p=1}^{P}\left(\prod_{j\in P_s(p)}\left(1-\underbrace{p_d^{[j]}(x)}_{=0}+\underbrace{p_d^{[j]}(x)}_{=0}p_{g^{[j]}}^{O,[j]}(x)\right)\right)s(x)\\
&=\mu s(x)
\end{aligned}
$$

The multi-sensor cross-term set in measurements $z^{[t_1]},...,z^{[t_M]}$ can be reformulated:

$$
\begin{aligned}
&\beta[g^{[1]},...,\delta_{z^{[t_1]}},...,\delta_{z^{[t_M]}},...,g^{[N]},h]\\
&=\begin{cases}
\lambda^{[t_1]}c^{[t_1]}(z^{[t_1]})+\mu s\left[hp_d^{[t_1]}L_{z^{[t_1]}}^{[t_1]}\prod_{j\neq t_1}\left(1-p_d^{[j]}+p_d^{[j]}p_{g^{[j]}}^{O,[j]}\right)\right] & (M=1)\\
\mu s\left[h\prod_{i=1}^{M}\left(p_d^{[t_i]}L_{z^{[t_i]}}^{[t_i]}\right)\prod_{j\notin\{t_1,...,t_M\}}\left(1-p_d^{[j]}+p_d^{[j]}p_{g^{[j]}}^{O,[j]}\right)\right] & (M>1)
\end{cases}\\
&=\begin{cases}
\lambda^{[t_1]}c^{[t_1]}(z^{[t_1]})+\mu\sum_{q=0}^{P}\int_{P_t(q)}h(x)p_d^{[t_1]}(x)L_{z^{[t_1]}}^{[t_1]}(x)\prod_{j\neq t_1}\left(1-p_d^{[j]}(x)+p_d^{[j]}(x)p_{g^{[j]}}^{O,[j]}(x)\right)s(x)dx & (M=1)\\
\mu\sum_{q=0}^{P}\int_{P_t(q)}h(x)\prod_{i=1}^{M}\left(p_d^{[t_i]}(x)L_{z^{[t_i]}}^{[t_i]}(x)\right)\prod_{j\notin\{t_1,...,t_M\}}\left(1-p_d^{[j]}(x)+p_d^{[j]}(x)p_{g^{[j]}}^{O,[j]}(x)\right)s(x)dx & (M>1)
\end{cases}
\end{aligned}
$$

If $\exists p \in [1\ P], \{t_1, ..., t_M\} \subseteq P_s(p)$ then:

$$\beta[g^{[1]}, ..., \delta_{z^{[t_1]}}, ..., \delta_{z^{[t_M]}}, ..., g^{[N]}, h]$$

$$= \begin{cases} \lambda^{[t_1]} c^{[t_1]}(z^{[t_1]}) \\ \quad + \mu \int_{P_t(p)} h(x) p_d^{[t_1]}(x) L_{z^{[t_1]}}^{[t_1]}(x) \prod_{\substack{j \in P_s(p) \\ j \neq t_1}} \left(1 - p_d^{[j]}(x) + p_d^{[j]}(x) p_{g^{[j]}}^{O,[j]}(x)\right) \prod_{j \notin P_s(p)} \left(1 - \underbrace{p_d^{[j]}(x)}_{=0} + \underbrace{p_d^{[j]}(x)}_{=0} p_{g^{[j]}}^{O,[j]}(x)\right) s(x)dx \\ \quad + \mu \sum_{q \neq p} \int_{P_t(q)} h(x) \underbrace{p_d^{[t_1]}(x)}_{=0} L_{z^{[t_1]}}^{[t_1]}(x) \prod_{j \neq t_1} \left(1 - p_d^{[j]}(x) + p_d^{[j]}(x) p_{g^{[j]}}^{O,[j]}(x)\right) s(x)dx \\ \mu \int_{P_t(p)} h(x) \prod_{i=1}^{M} \left(p_d^{[t_i]}(x) L_{z^{[t_i]}}^{[t_i]}(x)\right) \prod_{\substack{j \in P_s(p) \\ j \notin \{t_1, ..., t_M\}}} \left(1 - p_d^{[j]}(x) + p_d^{[j]}(x) p_{g^{[j]}}^{O,[j]}(x)\right) \prod_{j \notin P_s(p)} \left(1 - \underbrace{p_d^{[j]}(x)}_{=0} + \underbrace{p_d^{[j]}(x)}_{=0} p_{g^{[j]}}^{O,[j]}(x)\right) s \\ \quad + \mu \sum_{q \neq p} \int_{P_t(q)} h(x) \prod_{i=1}^{M} \left(\underbrace{p_d^{[t_i]}(x)}_{=0} L_{z^{[t_i]}}^{[t_i]}(x)\right) \prod_{j \notin \{t_1, ..., t_M\}} \left(1 - p_d^{[j]}(x) + p_d^{[j]}(x) p_{g^{[j]}}^{O,[j]}(x)\right) s(x)dx \end{cases}$$

$$= \begin{cases} \lambda^{[t_1]} c^{[t_1]}(z^{[t_1]}) + \mu \int_{P_t(p)} h(x) p_d^{[t_1]}(x) L_{z^{[t_1]}}^{[t_1]}(x) \prod_{\substack{j \in P_s(p) \\ j \neq t_1}} \left(1 - p_d^{[j]}(x) + p_d^{[j]}(x) p_{g^{[j]}}^{O,[j]}(x)\right) s(x)dx & (M = 1) \\ \mu \int_{P_t(p)} h(x) \prod_{i=1}^{M} \left(p_d^{[t_i]}(x) L_{z^{[t_i]}}^{[t_i]}(x)\right) \prod_{\substack{j \in P_s(p) \\ j \notin \{t_1, ..., t_M\}}} \left(1 - p_d^{[j]}(x) + p_d^{[j]}(x) p_{g^{[j]}}^{O,[j]}(x)\right) s(x)dx & (M > 1) \end{cases}$$

$$= \begin{cases} \lambda^{[t_1]} c^{[t_1]}(z^{[t_1]}) + \mu s \left[h p_d^{[t_1]} L_{z^{[t_1]}}^{[t_1]} \prod_{\substack{j \in P_s(p) \\ j \neq t_1}} \left(1 - p_d^{[j]} + p_d^{[j]} p_{g^{[j]}}^{O,[j]}\right) \right]_p & (M = 1) \\ \mu s \left[h \prod_{i=1}^{M} \left(p_d^{[t_i]} L_{z^{[t_i]}}^{[t_i]}\right) \prod_{\substack{j \in P_s(p) \\ j \notin \{t_1, ..., t_M\}}} \left(1 - p_d^{[j]} + p_d^{[j]} p_{g^{[j]}}^{O,[j]}\right) \right]_p & (M > 1) \end{cases}$$

$$= \beta_p[g^{[p_1]}, ..., \delta_{z^{[t_1]}}, ..., \delta_{z^{[t_M]}}, ..., g^{[p_{N_p}]}, h] \quad (\textit{using prop. } 5.1)$$

Otherwise $\exists p_1, p_2 \in [1\ P], p_1 \neq p_2, \exists i_1, i_2 \in [1\ M], i_1 \neq i_2, t_{i_1} \in P_s(p_1), t_{i_2} \in P_s(p_2)$ and therefore:

$$
\begin{aligned}
&\beta[g^{[1]}, ..., \delta_{z^{[t_1]}}, ..., \delta_{z^{[t_M]}}, ..., g^{[N]}, h] \\
&= \mu \int_{P_t(p_1)} h(x) \underbrace{p_d^{[t_{i_2}]}(x)}_{=0} L^{[t_{i_2}]}_{z^{[t_{i_2}]}}(x) \prod_{i \neq i_2} \left(p_d^{[t_i]}(x) L^{[t_i]}_{z^{[t_i]}}(x)\right) \prod_{j \notin \{t_1, ..., t_M\}} \left(1 - p_d^{[j]}(x) + p_d^{[j]}(x) p^{O,[j]}_{g^{[j]}}(x)\right) s(x) dx \\
&+ \mu \int_{P_t(p_2)} h(x) \underbrace{p_d^{[t_{i_1}]}(x)}_{=0} L^{[t_{i_1}]}_{z^{[t_{i_1}]}}(x) \prod_{i \neq i_1} \left(p_d^{[t_i]}(x) L^{[t_i]}_{z^{[t_i]}}(x)\right) \prod_{j \notin \{t_1, ..., t_M\}} \left(1 - p_d^{[j]}(x) + p_d^{[j]}(x) p^{O,[j]}_{g^{[j]}}(x)\right) s(x) dx \\
&+ \mu \sum_{\substack{q \neq p_1 \\ q \neq p_2}} \int_{P_t(q)} h(x) \underbrace{p_d^{[t_{i_1}]}(x)}_{=0} L^{[t_{i_1}]}_{z^{[t_{i_1}]}}(x) \prod_{i \neq i_1} \left(p_d^{[t_i]}(x) L^{[t_i]}_{z^{[t_i]}}(x)\right) \prod_{j \notin \{t_1, ..., t_M\}} \left(1 - p_d^{[j]}(x) + p_d^{[j]}(x) p^{O,[j]}_{g^{[j]}}(x)\right) s(x) dx \\
&= 0
\end{aligned}
$$

The bounded cross-term set in measurements $z^{[t_1]}, ..., z^{[t_M]}$ and target state x can be reformulated:

$$
\begin{aligned}
&\beta[g^{[1]}, ..., \delta_{z^{[t_1]}}, ..., \delta_{z^{[t_M]}}, ..., g^{[N]}, \delta_x] \\
&= \mu \prod_{i=1}^{M} \left(p_d^{[t_i]}(x) L^{[t_i]}_{z^{[t_i]}}(x)\right) \prod_{j \notin \{t_1, ..., t_M\}} \left(1 - p_d^{[j]}(x) + p_d^{[j]}(x) p^{O,[j]}_{g^{[j]}}(x)\right) s(x)
\end{aligned}
$$

If $\exists p \in [1\ P], \{t_1, ..., t_M\} \subseteq P_s(p), x \in P_t(p)$ then:

$$
\begin{aligned}
&\beta[g^{[1]}, ..., \delta_{z^{[t_1]}}, ..., \delta_{z^{[t_M]}}, ..., g^{[N]}, \delta_x] \\
&= \mu \prod_{i=1}^{M} \left(p_d^{[t_i]}(x) L^{[t_i]}_{z^{[t_i]}}(x)\right) \prod_{\substack{j \in P_s(p) \\ j \notin \{t_1, ..., t_M\}}} \left(1 - p_d^{[j]}(x) + p_d^{[j]}(x) p^{O,[j]}_{g^{[j]}}(x)\right) \prod_{j \notin P_s(p)} \left(1 - \underbrace{p_d^{[j]}(x)}_{=0} + \underbrace{p_d^{[j]}(x)}_{=0} p^{O,[j]}_{g^{[j]}}(x)\right) s(x) \\
&= \mu \prod_{i=1}^{M} \left(p_d^{[t_i]}(x) L^{[t_i]}_{z^{[t_i]}}(x)\right) \prod_{\substack{j \in P_s(p) \\ j \notin \{t_1, ..., t_M\}}} \left(1 - p_d^{[j]}(x) + p_d^{[j]}(x) p^{O,[j]}_{g^{[j]}}(x)\right) s(x) \\
&= \beta_p[g^{[p_1]}, ..., \delta_{z^{[t_1]}}, ..., \delta_{z^{[t_M]}}, ..., g^{[p_{N_p}]}, \delta_x] \quad (\textit{using prop. } 5.1)
\end{aligned}
$$

Otherwise, either $x \in P_t(0)$ or $\exists p_1, p_2 \in [1\ P], p_1 \neq p_2, \exists i_1 \in [1\ M], x \in P_t(p_1), t_{i_1} \in P_s(p_2)$ and in both cases $p_d^{[t_{i_1}]}(x) = 0$ so:

$$
\begin{aligned}
&\beta[g^{[1]}, ..., \delta_{z^{[t_1]}}, ..., \delta_{z^{[t_M]}}, ..., g^{[N]}, \delta_x] \\
&= \mu \underbrace{p_d^{[t_{i_1}]}(x)}_{=0} L^{[t_{i_1}]}_{z^{[t_{i_1}]}}(x) \prod_{i \neq i_1} \left(p_d^{[t_i]}(x) L^{[t_i]}_{z^{[t_i]}}(x)\right) \prod_{j \notin \{t_1, ..., t_M\}} \left(1 - p_d^{[j]}(x) + p_d^{[j]}(x) p^{O,[j]}_{g^{[j]}}(x)\right) s(x) \\
&= 0
\end{aligned}
$$

□

8.19 Proposition 6.2

Proof. The proof is trivial if $P = 1$ (i.e. the FOVs of all the sensors belongs to the same equivalence class). Moreover, the results from proposition 6.1 hold for any partition coarser than P_s. Thus, it is sufficient to prove that proposition 6.2 is true for $P = 2$. Indeed, applying the result with $P = 2$ successively on finer and finer partitions will yield the result for a partition with any number of elements, namely P_s.

Let $P = 2$ and let $x \in \mathcal{X}$ be a given state point. If $x \in P_t(0)$, i.e. if x does not lie in any FOV, then by using propositions 6.1 and 5.3 we have:

$$D_{k+1|k+1}(x) = \underbrace{\beta[\emptyset, \delta_x]}_{=\mu s(x)} + \frac{\sum\limits_{T \in \mathcal{T}(\check{Z}_N)} \sum\limits_{T_i \in T} \left(\underbrace{\beta[T_i, \delta_x]}_{=0} \prod\limits_{\substack{T_j \in T \\ T_j \neq T_i}} \beta[T_j, 1] \right)}{\sum\limits_{T \in \mathcal{T}(\check{Z}_N)} \prod\limits_{T_i \in T} \beta[T_i, 1]} = D_{k+1|k}(x)$$

Otherwise, since $P = 2$, either $x \in P_t(1)$ or $x \in P_t(2)$. Without loss of generality, it is assumed here that $x \in P_t(1)$. Let $T \in \mathcal{T}(\check{Z}_N)$ be any (non empty) combinational term and $T_i \in T$ any tuple belonging to T. Then, according to proposition 6.1:

$$\beta[T_i, h] = \begin{cases} \beta_1[T_i, h] & (T_i \in \check{Z}^{(1)}_{N_1}) \\ \beta_2[T_i, h] & (T_i \in \check{Z}^{(2)}_{N_2}) \\ 0 & (otherwise) \end{cases}$$

Therefore we can write:

$$\sum_{T \in \mathcal{T}(\check{Z}_N)} \prod_{T_i \in T} \beta[T_i, 1] = \sum_{\substack{T \in \mathcal{T}(\check{Z}_N) \\ T = T^{(1)} \cup T^{(2)} \\ T^{(1)} \subseteq \check{Z}^{(1)}_{N_1} \\ T^{(2)} \subseteq \check{Z}^{(2)}_{N_2}}} \left(\prod_{T_i \in T^{(1)}} \beta_1[T_i, 1] \right) \left(\prod_{T_i \in T^{(2)}} \beta_2[T_i, 1] \right)$$

Which gives, using equations (45) and (55):

$$= \sum_{\substack{T = T^{(1)} \cup T^{(2)} \\ T^{(1)} \in \mathcal{T}(\check{Z}^{(1)}_{N_1}) \\ T^{(2)} \in \mathcal{T}(\check{Z}^{(2)}_{N_2})}} \left(\prod_{T_i \in T^{(1)}} \beta_1[T_i, 1] \right) \left(\prod_{T_i \in T^{(2)}} \beta_2[T_i, 1] \right)$$

$$= \left(\sum_{T \in \mathcal{T}(\check{Z}^{(1)}_{N_1})} \prod_{T_i \in T} \beta_1[T_i, 1] \right) \left(\sum_{T \in \mathcal{T}(\check{Z}^{(2)}_{N_2})} \prod_{T_i \in T} \beta_2[T_i, 1] \right)$$

Likewise, the numerator in data update equation (51) can be simplified:

$$
\begin{aligned}
&\sum_{T\in\mathcal{T}(\check{Z}_N)}\sum_{T_i\in T}\left(\beta[T_i,\delta_x]\prod_{\substack{T_j\in T\\T_j\neq T_i}}\beta[T_j,1]\right)\\
&=\sum_{\substack{T\in\mathcal{T}(\check{Z}_N)\\T=T^{(1)}\cup T^{(2)}\\T^{(1)}\subseteq\check{Z}^{(1)}_{N_1}\\T^{(2)}\subseteq\check{Z}^{(2)}_{N_2}}}\sum_{T_i\in T}\left(\beta[T_i,\delta_x]\left(\prod_{\substack{T_j\in T^{(1)}\\T_j\neq T_i}}\beta_1[T_j,1]\right)\left(\prod_{\substack{T_k\in T^{(2)}\\T_k\neq T_i}}\beta_2[T_k,1]\right)\right)\\
&=\sum_{\substack{T=T^{(1)}\cup T^{(2)}\\T^{(1)}\in\mathcal{T}(\check{Z}^{(1)}_{N_1})\\T^{(2)}\in\mathcal{T}(\check{Z}^{(2)}_{N_2})}}\sum_{T_i\in T^{(1)}}\left(\beta_1[T_i,\delta_x]\prod_{\substack{T_j\in T^{(1)}\\T_j\neq T_i}}\beta_1[T_j,1]\right)\left(\prod_{T_i\in T^{(2)}}\beta_2[T_i,1]\right)+\\
&\sum_{\substack{T=T^{(1)}\cup T^{(2)}\\T^{(1)}\in\mathcal{T}(\check{Z}^{(1)}_{N_1})\\T^{(2)}\in\mathcal{T}(\check{Z}^{(2)}_{N_2})}}\sum_{T_i\in T^{(2)}}\left(\prod_{T_j\in T^{(1)}}\beta_1[T_j,1]\right)\left(\underbrace{\beta_2[T_i,\delta_x]}_{\substack{=0\\(x\in P_t(1))}}\prod_{\substack{T_k\in T^{(2)}\\T_k\neq T_i}}\beta_2[T_k,1]\right)\\
&=\left(\sum_{T\in\mathcal{T}(\check{Z}^{(1)}_{N_1})}\sum_{T_i\in T}\left(\beta_1[T_i,\delta_x]\prod_{\substack{T_j\in T\\T_j\neq T_i}}\beta_1[T_j,1]\right)\right)\left(\sum_{T\in\mathcal{T}(\check{Z}^{(2)}_{N_2})}\prod_{T_i\in T}\beta_2[T_i,1]\right)
\end{aligned}
$$

Finally, replacing in equation (51) the simplified expressions of the numerator and denominator above gives the result:

$$
\begin{aligned}
D_{k+1|k+1}(x)&=\beta[\emptyset,\delta_x]+\frac{\left(\sum_{T\in\mathcal{T}(\check{Z}^{(1)}_{N_1})}\sum_{T_i\in T}\left(\beta_1[T_i,\delta_x]\prod_{\substack{T_j\in T\\T_j\neq T_i}}\beta_1[T_j,1]\right)\right)\left(\sum_{T\in\mathcal{T}(\check{Z}^{(2)}_{N_2})}\prod_{T_i\in T}\beta_2[T_i,1]\right)}{\left(\sum_{T\in\mathcal{T}(\check{Z}^{(1)}_{N_1})}\prod_{T_i\in T}\beta_1[T_i,1]\right)\left(\sum_{T\in\mathcal{T}(\check{Z}^{(2)}_{N_2})}\prod_{T_i\in T}\beta_2[T_i,1]\right)}\\
&=\beta_1[\emptyset,\delta_x]+\frac{\sum_{T\in\mathcal{T}(\check{Z}^{(1)}_{N_1})}\sum_{T_i\in T}\left(\beta_1[T_i,\delta_x]\prod_{\substack{T_j\in T\\T_j\neq T_i}}\beta_1[T_j,1]\right)}{\sum_{T\in\mathcal{T}(\check{Z}^{(1)}_{N_1})}\prod_{T_i\in T}\beta_1[T_i,1]}
\end{aligned}
$$

$\square$

8.20 Proposition 6.4

Proof. Let us prove by induction that for any sensor i in $[1\ N]$ and any x in $\mathcal{X}$, we have:

$$\mu^{[i]}s^{[i]}(x) = \left[\prod_{k\leqslant i}\left(1-p_d^{[k]}(x)\right) + \sum_{1\leqslant k^1\leqslant i}\left(\sum_{j_1=1}^{m[k^1]}\frac{\prod_{k\leqslant i,k\neq k^1}\left(1-p_d^{[k]}(x)\right).p_d^{[k^1]}(x)L^{[k^1]}_{z^{[k^1]}_{j_1}}(x)}{Den_M^{[k^1]}(z^{[k^1]}_{j_1})}\right)\right.$$
$$\left.+\cdots+\sum_{1\leqslant k^1\leqslant\cdots\leqslant k^i\leqslant i}\left(\sum_{j_1=1}^{m[k^1]}\cdots\sum_{j_i=1}^{m[k^i]}\frac{p_d^{[k^1]}(x)L^{[k^1]}_{z^{[k^1]}_{j_1}}(x)\cdots p_d^{[k^i]}(x)L^{[k^i]}_{z^{[k^i]}_{j_i}}(x)}{Den_M^{[k^1]}(z^{[k^1]}_{j_1})\cdots Den_M^{[k^i]}(z^{[k^i]}_{j_i})}\right)\right]\mu s(x)$$

The base case is trivial, using definition 6.4 we have:

$$\begin{aligned}\mu^{[1]}s^{[1]}(x) &= C_M^{[1]}(Z_{k+1}^{[1]}|x)\mu s(x)\\ &= \left[1-p_d^{[1]}+\sum_{z\in Z_{k+1}^{[1]}}\frac{p_d^{[1]}(x)L_z^{[1]}(x)}{\lambda^{[1]}c^{[1]}(z)+\mu s[p_d^{[1]}L_z^{[1]}]}\right]\mu s(x)\\ &= \left[1-p_d^{[1]}+\sum_{j_1=1}^{m[1]}\frac{p_d^{[1]}(x)L^{[1]}_{z^{[1]}_{j_1}}(x)}{Den_M^{[1]}(z^{[1]}_{j_1})}\right]\mu s(x)\end{aligned}$$

Assuming that the induction is true up to step $i \in [1\ N-1]$, let us prove it at step $i+1$. Once again using definition 6.4, we have:

$$\mu^{[i+1]}s^{[i+1]}(x)$$

$$= C_M^{[i+1]}(Z_{k+1}^{[i+1s]}|x)\mu^{[i]}s^{[i]}(x)$$

$$= \left[1 - p_d^{[i+1]}(x) + \sum_{j_{i+1}=1}^{m[i+1]} \frac{p_d^{[i+1]}(x) L_{z_{j_{i+1}}^{[i+1]}}^{[i+1]}(x)}{\underbrace{\lambda^{[i+1]} c^{[i+1]}(z_{j_{i+1}}^{[i+1]}) + \mu^{[i]} s^{[i]} \left[p_d^{[i+1]} L_{z_{j_{i+1}}^{[i+1]}}^{[i+1]} \right]}_{Den_M^{[i+1]}(z_{j_{i+1}}^{[i+1]})}} \right] \mu^{[i]} s^{[i]}(x)$$

$$= \left[1 - p_d^{[i+1]}(x) + \sum_{j_{i+1}=1}^{m[i+1]} \frac{p_d^{[i+1]}(x) L_{z_{j_{i+1}}^{[i+1]}}^{[i+1]}(x)}{Den_M^{[i+1]}(z_{j_{i+1}}^{[i+1]})} \right] \cdot \left[\prod_{k \leqslant i} \left(1 - p_d^{[k]}(x)\right) + \sum_{1 \leqslant k^1 \leqslant i} \left(\sum_{j_1=1}^{m[k^1]} \frac{\prod_{k \leqslant i, k \neq k^1} \left(1 - p_d^{[k]}(x)\right) . p_d^{[k^1]}(x) L_{z_{j_1}^{[k^1]}}^{[k^1]}(x)}{Den_M^{[k^1]}(z_{j_1}^{[k^1]})} \right) \right.$$

$$\left. + \cdots + \sum_{1 \leqslant k^1 \leqslant \ldots \leqslant k^i \leqslant i} \left(\sum_{j_1=1}^{m[k^1]} \cdots \sum_{j_i=1}^{m[k^i]} \frac{p_d^{[k^1]}(x) L_{z_{j_1}^{[k^1]}}^{[k^1]}(x) \cdots p_d^{[k^i]}(x) L_{z_{j_i}^{[k^i]}}^{[k^i]}(x)}{Den_M^{[k^1]}(z_{j_1}^{[k^1]}) \cdots Den_M^{[k^i]}(z_{j_i}^{[k^i]})} \right) \right] \mu s(x)$$

$$= \left[\prod_{k \leqslant i+1} \left(1 - p_d^{[k]}(x)\right) + \sum_{1 \leqslant k^1 < i+1} \left(\sum_{j_1=1}^{m[k^1]} \frac{\prod_{k \leqslant i+1, k \neq k^1} \left(1 - p_d^{[k]}(x)\right) . p_d^{[k^1]}(x) L_{z_{j_1}^{[k^1]}}^{[k^1]}(x)}{Den_M^{[k^1]}(z_{j_1}^{[k^1]})} \right) \right.$$

$$+ \cdots + \sum_{1 \leqslant k^1 \leqslant \ldots \leqslant k^i < i+1} \left(\sum_{j_1=1}^{m[k^1]} \cdots \sum_{j_i=1}^{m[k^i]} \frac{\prod_{k \leqslant i+1, k \neq k^1, \ldots, k \neq k^i} \left(1 - p_d^{[k]}(x)\right) . p_d^{[k^1]}(x) L_{z_{j_1}^{[k^1]}}^{[k^1]}(x) \cdots p_d^{[k^i]}(x) L_{z_{j_i}^{[k^i]}}^{[k^i]}(x)}{Den_M^{[k^1]}(z_{j_1}^{[k^1]}) \cdots Den_M^{[k^i]}(z_{j_i}^{[k^i]})} \right)$$

$$+ \sum_{1 \leqslant k^1 = i+1} \left(\sum_{j_1=1}^{m[k^1]} \frac{\prod_{k \leqslant i+1, k \neq k^1} \left(1 - p_d^{[k]}(x)\right) . p_d^{[k^1]}(x) L_{z_{j_1}^{[k^1]}}^{[k^1]}(x)}{Den_M^{[k^1]}(z_{j_1}^{[k^1]})} \right)$$

$$\left. + \cdots + \sum_{1 \leqslant k^1 \leqslant \ldots \leqslant k^i \leqslant k^{i+1} = i+1} \left(\sum_{j_1=1}^{m[k^1]} \cdots \sum_{j_i=1}^{m[k^{i+1}]} \frac{p_d^{[k^1]}(x) L_{z_{j_1}^{[k^1]}}^{[k^1]}(x) \cdots p_d^{[k^i]}(x) L_{z_{j_{i+1}}^{[k^{i+1}]}}^{[k^{i+1}]}(x)}{Den_M^{[k^1]}(z_{j_1}^{[k^1]}) \cdots Den_M^{[k^{i+1}]}(z_{j_{i+1}}^{[k^{i+1}]})} \right) \right] \mu s(x)$$

$$= \left[\prod_{k \leqslant i+1} \left(1 - p_d^{[k]}(x)\right) + \sum_{1 \leqslant k^1 \leqslant i+1} \left(\sum_{j_1=1}^{m[k^1]} \frac{\prod_{k \leqslant i+1, k \neq k^1} \left(1 - p_d^{[k]}(x)\right) . p_d^{[k^1]}(x) L_{z_{j_1}^{[k^1]}}^{[k^1]}(x)}{Den_M^{[k^1]}(z_{j_1}^{[k^1]})} \right) \right.$$

$$\left. + \cdots + \sum_{1 \leqslant k^1 \leqslant \ldots \leqslant k^{i+1} \leqslant i+1} \left(\sum_{j_1=1}^{m[k^1]} \cdots \sum_{j_{i+1}=1}^{m[k^{i+1}]} \frac{p_d^{[k^1]}(x) L_{z_{j_1}^{[k^1]}}^{[k^1]}(x) \cdots p_d^{[k^{i+1}]}(x) L_{z_{j_{i+1}}^{[k^{i+1}]}}^{[k^{i+1}]}(x)}{Den_M^{[k^1]}(z_{j_1}^{[k^1]}) \cdots Den_M^{[k^{i+1}]}(z_{j_{i+1}}^{[k^{i+1}]})} \right) \right] \mu s(x)$$

Therefore, the induction is true up to step N. Then, using definition 6.4 once more, we have:

$$
\begin{aligned}
D_{k+1|k+1}(x) &\simeq \mu^{[N]} s^{[N]}(x) \\
&\simeq \Bigg[\prod_{k \leqslant N} \left(1 - p_d^{[k]}(x)\right) + \sum_{1 \leqslant k^1 \leqslant N} \left(\sum_{j_1=1}^{m[k^1]} \frac{\prod_{k \neq k^1} \left(1 - p_d^{[k]}(x)\right) . p_d^{[k^1]}(x) L^{[k^1]}_{z^{[k^1]}_{j_1}}(x)}{Den_M^{[k^1]}(z^{[k^1]}_{j_1})} \right) \\
&\quad + \cdots + \sum_{1 \leqslant k^1 \leqslant \ldots \leqslant k^N \leqslant N} \left(\sum_{j_1=1}^{m[k^1]} \cdots \sum_{j_N=1}^{m[k^N]} \frac{p_d^{[k^1]}(x) L^{[k^1]}_{z^{[k^1]}_{j_1}}(x) \cdots p_d^{[k^N]}(x) L^{[k^N]}_{z^{[k^N]}_{j_N}}(x)}{Den_M^{[k^1]}(z^{[k^1]}_{j_1}) \cdots Den_M^{[k^N]}(z^{[k^N]}_{j_N})} \right) \Bigg] D_{k+1|k}(x)
\end{aligned}
$$

$\square$

References

[1] R. P. S. Mahler. Multitarget bayes filtering via first-order multitarget moments. *IEEE Transactions on Aerospace and Electronic Systems*, 39(4):1152–1178, October 2003.

[2] R. P. S. Mahler. Objective functions for bayesian control-theoretic sensor management, i: Multitarget first-moment approximation. In *Aerospace Conference, 2003. Proceedings. 2003 IEEE*, volume 4, pages 1905–1923, Mar 2003.

[3] R. P. S. Mahler. Phd filters of higher order in target number. *IEEE Transactions on Aerospace and Electronic Systems*, 43(4):1523–1543, October 2007.

[4] R. P. S. Mahler. The multisensor phd filter, i: General solution via multitarget calculus. In *Signal Processing, Sensor Fusion, and Target Recognition XVIII, Proceedings of SPIE*, April 2009.

[5] R. P. S. Mahler. Approximate multisensor cphd and phd filters. In *Information Fusion, 2010, Proceedings of the 13th International Conference on*, July 2010.

[6] B-T. Vo. *Random Finite Sets in Multi-Object Filtering.* PhD thesis, The University of Western Australia, October 2008.

Centre de recherche INRIA Lille – Nord Europe
Parc Scientifique de la Haute Borne - 40, avenue Halley - 59650 Villeneuve d'Ascq (France)

Centre de recherche INRIA Bordeaux – Sud Ouest : Domaine Universitaire - 351, cours de la Libération - 33405 Talence Cedex
Centre de recherche INRIA Grenoble – Rhône-Alpes : 655, avenue de l'Europe - 38334 Montbonnot Saint-Ismier
Centre de recherche INRIA Nancy – Grand Est : LORIA, Technopôle de Nancy-Brabois - Campus scientifique
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex
Centre de recherche INRIA Paris – Rocquencourt : Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex
Centre de recherche INRIA Rennes – Bretagne Atlantique : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex
Centre de recherche INRIA Saclay – Île-de-France : Parc Orsay Université - ZAC des Vignes : 4, rue Jacques Monod - 91893 Orsay Cedex
Centre de recherche INRIA Sophia Antipolis – Méditerranée : 2004, route des Lucioles - BP 93 - 06902 Sophia Antipolis Cedex

Éditeur
INRIA - Domaine de Voluceau - Rocquencourt, BP 105 - 78153 Le Chesnay Cedex (France)
http://www.inria.fr
ISSN 0249-6399